DFSMS/MVS Version 1 Release 5

DFSMSdfp Storage Administration Reference

SC26-4920-05

DFSMS/MVS Version 1 Release 5

DFSMSdfp Storage Administration Reference

SC26-4920-05

Note!

Before using this information and the product it supports, be sure to read the general information under "Notices" on page xiii.

Sixth Edition (March 1999)

This edition applies to Version 1 Release 5 of DFSMS/MVS (5695-DF1), Release 7 of OS/390 (5647-A01), and any subsequent releases until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

A form for readers' comments appears at the back of this publication. If the form has been removed, address your comments to:

International Business Machines Corporation
RCF Processing Department
G26/050
5600 Cottle Road
SAN JOSE, CA 95193-0001
U.S.A.

Or you can send your comments electronically to starpubs@vnet.ibm.com.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Figures . . . ix

Notices . . . xiii
Programming Interface Information . . . xiii
Trademarks . . . xiv

About This Book . . . xv
Required Product Knowledge . . . xv
How to Tell if this Book is Current . . . xvi
Referenced Publications . . . xvi
References to Product Names Used in DFSMS/MVS Publications . . . xviii
Notational Conventions . . . xix

Summary of Changes . . . xxi
Sixth Edition, March 1999 . . . xxi
Fifth Edition, June 1997 . . . xxiii
Fourth Edition, September 1996 . . . xxv

Chapter 1. Introducing the Storage Management Subsystem . . . 1
Defining a Storage Management Policy . . . 1
Using the Interactive Storage Management Facility (ISMF) . . . 2
Implementing the Storage Management Subsystem . . . 4
Running SMS in a Parallel Sysplex Environment . . . 5
  Basic Terms and Definitions . . . 5
  System Grouping Concept . . . 6
Running with Mixed Levels of DSMS/MVS . . . 7

Chapter 2. Preparing for the Storage Management Subsystem . . . 9
Allocating Control Data Sets . . . 9
  Calculating the Size of Storage and Active Control Data Sets . . . 11
  Calculating the Size of a COMMDS . . . 12
  Selecting Volumes for Control Data Sets . . . 13
  Allocating an SCDS . . . 13
  Allocating an ACDS . . . 14
  Allocating a COMMDS . . . 14
Modifying the SYS1.PARMLIB Data Set . . . 15
  Initializing SMS through the IGDSMSxx Member . . . 15
  Defining SMS to MVS through IEFSSNxx . . . 23
  Starting the SMS Address Space . . . 25
Accessing the Storage Administrator Primary Option Menu . . . 25
Planning for VSAM Record-Level Sharing . . . 26

Chapter 3. Creating the Base Configuration . . . 27
Planning the Base Configuration . . . 27
Defining the Base Configuration . . . 27
  Specifying the Default Management Class . . . 30
  Specifying the Default Unit . . . 31
  Specifying the Default Device Geometry . . . 31
  Specifying Systems and System Groups in the SMS Complex . . . 31
Defining the Base Configuration for VSAM Record-Level Sharing . . . 33
  Using Cache Structures . . . 33
  Defining Cache Sets . . . 33
Completing the Base Configuration . . . 35

Chapter 4. Defining Storage Groups 37
Understanding Storage Groups 37
DASD Storage Groups. 38
Tracking DASD Volume Status for Data Sets 38
Object and Object Backup Storage Groups 39
OAM Collection Names 41
Tape Storage Groups 42
Planning Storage Groups for Data Sets 42
Planning Storage Groups for OAM Object Collections 44
Defining Storage Group Attributes 44
Defining a VIO Storage Group 46
Defining a Dummy Storage Group 49
Defining a Pool Storage Group 51
Assigning DASD Storage Groups to Data Sets 56
Defining an Object Storage Group 56
Defining the Object Backup Storage Group 58
Assigning an OAM Object Collection to a Storage Group 61
Defining a Tape Storage Group 62
Defining Additional Storage Groups 63
Listing Volumes in a Storage Group 64

Chapter 5. Defining Management Classes 65
Understanding Management Classes 65
Default Management Class 65
OAM Management Classes 66
Describing Management Classes 66
Planning Management Classes 66
Defining Management Class Attributes 67
Defining Management Class Expiration Attributes 69
Defining Management Class Migration Attributes 72
Defining Management Class Backup Attributes 75
Defining Object Class Backup Attributes 78
Defining Aggregate Backup Attributes 79
Assigning Management Classes 80
Defining Additional Management Classes 81

Chapter 6. Defining Storage Classes 83
Understanding Storage Classes 83
Storage Classes for Data Sets 83
Storage Classes for Objects. 84
Planning Storage Classes 84
Defining Storage Class Attributes 85
Defining Performance Objectives 86
Defining Availability 93
Defining Accessibility 93
Defining Guaranteed Space 95
Defining Guaranteed Synchronous Write 97
Defining Use of the Coupling Facility for VSAM Record-Level Sharing . . . 97
Assigning Storage Classes 98
Defining Additional Storage Classes 99
Understanding Volume Selection 99
How SMS Selects Volumes for Data Set Allocation 100
Conventional Volume Selection 105
Striping Volume Selection 110
Some Possible Reasons for Failure in Volume Selection 112

Chapter 7. Defining Data Classes 115
Understanding Data Classes 115
Planning Data Classes. 115
Defining Data Class Attributes 116
Defining Record and Space Attributes for Data Class 117
Defining Volume and VSAM Attributes for Data Class 120
Defining Data Set Type Attributes for Data Class 123
Specifying Attributes for Data Set Reuse, Loading, and Spanned Record Format. 125
Specifying Attributes for Backup-While-Open (BWO) and Recovery 126
Specifying Attributes to Handle Space Constraints During Allocation 127
Assigning Data Classes 129
Specifying Data Classes outside ACS Routines 129
Processing of Data Class Attributes in JCL 129
Defining Additional Data Classes 130

Chapter 8. Defining Aggregate Groups 131
Understanding Aggregate Groups. 131
Planning Aggregate Groups 132
Defining Aggregate Groups 132
Defining Aggregate Group Attributes. 133
Editing Aggregate Group Attributes 134
Backing up and Recovering an Aggregate Group 137
Defining Additional Aggregate Groups 137

Chapter 9. Defining ACS Routines 139
Understanding ACS Routines 139
Using ACS Routines for Data Sets Created by DFSMS/MVS Network File System and Distributed FileManager/MVS. 140
Restrictions on Using ACS Routines. 140
Creating ACS Routines 141
Translating ACS Routines 142
Browsing the Results of a Translation 143
BROWSING the Results of an Unsuccessful Translation 145
Validating ACS Routines or an Entire SCDS 146
Validating an ACS Routine 146
Validating an Entire SCDS 149
Translating and Validating in a Multiprocessor Environment 150
Testing ACS Routines 151
ACS Test Usage 151
Creating ACS Test Cases 152
Running ACS Test Cases. 155
ACS Routines Invoked for Copying and Importing Data Sets. 156
ACS Routines Invoked for Restoring, Recalling, Recovering, and Converting Data Sets 158
ACS Routine Environments 160
JCL DD Statement (Batch), and Dynamic Allocation 160
Volume Reference 160
Data Set Stacking 162
Tape Management System Support 164
Access Method Services 165
DFSMShsm. 166
DFSMSdss 166
ISMF 166
OAM 167
Processing of SMS Classes and Storage Groups 167

Displaying ACS Object Information 168
Deleting an ACS Object from an SCDS 169

Chapter 10. Activating Storage Management Subsystem Configurations . . 171
Manually Activating the First Storage Management Subsystem Configuration . . 171
Step One: IPL Each System in the SMS Complex 171
Step Two: Prepare One System 171
Step Three: Activate the Configuration from One System 171
Step Four: Activate SMS on the Other Systems 172
Automatically Activating a Storage Management Subsystem Configuration . . . 172
Changing Storage Management Subsystem Parameters 173
SETSMS Operator Command 173
Considerations when Changing Storage Management Subsystem Configurations 177
OAM Considerations when Changing SCDSs 178

Chapter 11. Maintaining the Storage Management Subsystem 179
Displaying SMS and OAM Information 179
Displaying Information about the Active Configuration 179
Displaying SMS TRACE Information 180
Displaying Storage Group Status Using the DISPLAY SMS Command . . . 181
Displaying OAM Status in a Parallel Sysplex 183
Displaying the Caching Statistics 184
Displaying Storage Group Status Using ISMF 184
Displaying Volume Status Using ISMF 186
Displaying Volumes Using the DISPLAY SMS Command 187
Displaying the Status of a Device 189
Listing SMS Classes, Aggregate Groups, Storage Groups, and Libraries Using ISMF 190
Listing with View and Sort 190
Listing Data Classes 191
Listing Storage Groups, Management Classes, Storage Classes, Aggregate Groups, Libraries and Drives 194
Altering Data Set Associations 195
Altering SMS Classes, Aggregate Groups, Storage Groups, Libraries, Volumes, and Drives 195
Altering Storage Groups 196
Redefining Volumes among Storage Groups 196
Altering Management Classes 197
Altering Storage Classes 198
Altering Data Classes 199
Altering Aggregate Groups 201
Copying SMS Classes, Storage Groups, and Aggregate Groups 204
Deleting DASD Volumes from the System 205
Deleting DASD Volumes from Storage Groups 205
Deleting Storage Groups 206
Deleting or Moving System-Managed Data Sets from the Storage Group . . 206
Preventing Job Failures 207
Deleting SMS Classes 207
Converting Volumes to SMS 208

Chapter 12. Recovering Storage Management Subsystem Information . . 209
Recovering Control Data Sets 209
Recovering an SCDS 209
Recovering an ACDS 209
Recovering a COMMDS 210

Notes on Recovering Control Data Sets 210
Recovering from a Systems Failure in the SMS Complex 211
Recovering the SMS Address Space 211
Canceling the SMS Address Space 211

Chapter 13. Protecting the Storage Management Subsystem 213
Identifying the Resource Owner and Extracting the Default Classes 213
Protecting ISMF Functions 213
Locating Module Names for ISMF Applications 215
Locating Module Names for ISMF Functions. 215
Locating Module Names for Line Operators and Commands 217
Protecting Modules 220
Storage Administration (STGADMIN) Profiles in the FACILITY Class 221
  Command and Keyword Related Profiles 221
  Authority to Activate a Storage Management Subsystem Configuration . . . 224
  Storage Administrator Facility Class Profiles for DFSMSdss 224
FIELD Resource Class 226
How Authorization and Protection of Classes Differ 226
  Authorizing the Storage and Management Classes 227
  Protecting the Control Data Sets 227
  Authorizing for TSO and ISPF 227
  Storage Administrator Authorization 227

Chapter 14. Administering VSAM Record-Level Sharing 229
Preparing for VSAM Record-Level Sharing 229
  Determining Hardware Requirements 230
  Understanding the Product Environment for VSAM RLS 230
  Determining Which Applications Can Use VSAM RLS 231
  Ensuring Same Systems Connectivity 232
  Planning for Availability 233
  Defining Sharing Control Data Sets 233
  Defining CF Cache Structures 235
  Defining the CF Lock Structure 237
  Modifying the SYS1.PARMLIB IGDSMSxx Member 241
  Establishing Authorization for VSAM RLS 241
Defining CF Cache Structures in the SMS Base Configuration 241
Defining Storage Classes for VSAM RLS 244
Defining VSAM RLS Attributes in Data Classes 246
Activating VSAM RLS 246
  Enabling VSAM RLS Processing 246
  Enabling a Data Set for VSAM RLS Processing 247
Monitoring the Coupling Facility for VSAM RLS 247
  Displaying CF Information 247
  Changing the Size of Coupling Facility Cache Structures 249
  Changing the Size of the Coupling Facility Lock Structure. 250
  Changing IGDSMSxx Parameters to Support the Coupling Facility 250
  Changing the State of Coupling Facility Cache Structures and Volumes . . 252
  Selecting Data Sets for Coupling Facility Statistical Monitoring 252
Recovering VSAM RLS Processing 253
  Recovering the CF Lock Structure 253
  Recovering a CF Cache Structure 253
  Recovering the SMSVSAM Server Address Space 254
  Recovering a Shared Control Data Set. 254
Falling Back from VSAM RLS Processing. 254
  Fallback Rules and Considerations 255
  Fallback Procedure 255

Chapter 15. ACS Language Reference 257
Constants 257
Simple Mask Rules 258
Simple Mask Examples 258
Data Set Mask Rules 258
Data Set Mask Examples. 259
Read-Write Variables 259
Using Read-Write Variables with Volume Reference (VOL=REF) 260
Using Read-Write Variables with Data Set Stacking 260
Read-Only Variables 261
Using Read-Only Variables 269
Constraints When Using Read-Only Variables 271
Comparison Operators. 272
Comparison Rules 273
Boolean Expressions 273
Special Functions 274
Determining Distributed FileManager/MVS Data Set Creation Requests. . . 275
Statements 276
PROC Statement. 277
FILTLIST Statement. 278
SET Statement 279
DO Statement 280
IF Statement 280
SELECT Statement 281
EXIT Statement 283
WRITE Statement 283
END Statement 284
Sample ACS Routine 284

Appendix A. ISMF Command and Line Operator Reference Summary. . . 287

Appendix B. SETCACHE Functions and Device Information 295
Tasks That Maintain the Media Characteristics of a Volume 295
Ways to Modify DASD Storage Control Characteristics 296
Modifying Caching Characteristics 297
Modifying Duplexing Characteristics 298
Modifying Destaging Characteristics 298
Submitting Jobs 298
Ways to Provide Remote Authorization Codes 299

Appendix C. Collecting Space and Capacity Planning Information 301
Using the Data Collection Application 301

Abbreviations 307

Glossary 311

Index 323

Readers' Comments — We'd Like to Hear from You. 335

Figures

1. ISMF Primary Option Menu for Storage Administrators 3
2. Activating an SMS Configuration 4
3. System Grouping Concept 7
4. Relationship among SCDSs and ACDSs in an Installation 11
5. Keyword Syntax to Define SMS in IEFSSNxx 23
6. Positional Syntax to Define SMS in IEFSSNxx 24
7. CDS Application Selection 28
8. SCDS Base Define panel 29
9. SCDS Base Define panel 30
10. CDS Application Selection for VSAM RLS 34
11. Cache Set Update Panel for VSAM RLS 34
12. Formula for Deriving Optimum Number of Stripes 43
13. Defining a Storage Group 45
14. Defining VIO Storage Group Attributes 47
15. Defining Dummy Storage Group Attributes 49
16. Defining Dummy Storage Group Volume Serial Numbers 50
17. Defining Pool Storage Group Attributes 51
18. Defining Pool Storage Group System Status 54
19. Defining Volume System Status 55
20. Defining Object Storage Group Attributes 56
21. Defining Object Backup Storage Group Attributes 59
22. Defining Object or Object Backup Storage Group System Status 60
23. Defining Tape Storage Group Attributes. 62
24. Defining Tape Storage Group System Status. 63
25. Management Class Application Selection Panel for Storage Administrators, Page 1 of 2 68
26. Management Class Application Selection Panel for Storage Administrators, Page 2 of 2 69
27. Defining Management Class Expiration Attributes 70
28. Defining Management Class Migration Attributes 72
29. Defining Management Class Backup Attributes 75
30. Defining Object Class Backup Attributes 78
31. Defining Aggregate Backup Attributes 79
32. Defining a Storage Class 85
33. Defining Storage Class Attributes, Page 1 of 2 86
34. MSR Capabilities 88
35. Example of JCL to Allocate a Multivolume Data Set On SMS-Selected Volumes 96
36. Allocation of Primary and Secondary Space for Multivolume Data Sets . . 96
37. Example of JCL Preallocating Space for a Multivolume Data Set 97
38. Defining Storage Class Attributes - Page 2 98
39. How SMS Selects Volumes for Data Set Allocation (Part 1) 101
40. How SMS Selects Volumes for Data Set Allocation (Part 2) 102
41. How SMS Selects Volumes for Data Set Allocation (Part 3) 103
42. How SMS Selects Volumes for Data Set Allocation (Part 4) 104
43. How SMS Selects Volumes for Data Set Allocation (Part 5) 105
44. Defining a Data Class 116
45. Defining Data Class Record and Space Attributes 117
46. Defining Data Class Volume and VSAM Attributes 120
47. Defining Data Class Type and Addressability. 123
48. Aggregate Group Application Selection 133
49. Aggregate Group Define, Page 1 of 2 134
50. Aggregate Group Define, Page 2 of 2 135

51. Editing an Aggregate Group Selection Data Set- Example 136
52. Writing an ACS Routine 141
53. Invoking the ISPF/PDF Editor 142
54. Translating an ACS Routine 143
55. Browsing the Results of an ACS Translation 144
56. Generating an ACS Translation Listing 145
57. Browsing the Results of an Unsuccessful ACS Translation. 146
58. Validating an ACS Routine 147
59. Checking for Deleted Storage Classes 148
60. Browsing the Results of an ACS Routine Validation 148
61. Validating an SCDS 149
62. ACS Test Case Selection panel 152
63. Creating ACS Test Cases, Page 1 of 4 153
64. Creating ACS Test Cases, Page 2 of 4 153
65. Creating ACS Test Cases, Page 3 of 4 154
66. Creating ACS Test Cases, Page 4 of 4 154
67. Testing an ACS Routine 155
68. Results of Testing an ACS Routine 156
69. Example of REF=ST Values when Using VOL=REF 162
70. Example of REF=NS Values when Using VOL=REF 162
71. Example of Failing VOL=REF Values That Are Not Valid 162
72. Example of a Storage Class ACS Routine for Read-Only Variables 164
73. Example of a Storage Group ACS Routine for Read-Only Variables 164
74. Displaying ACS Object Information 168
75. Deleting ACS Objects 169
76. Displaying Information about the Active Configuration 180
77. Displaying Trace Information. 181
78. DISPLAY SMS,STORGRP Command Syntax 181
79. Displaying Storage Group Status Information 182
80. Displaying Storage Group Volume Status Information 183
81. Displaying OAM Status in a Parallel Sysplex. 184
82. Displaying Cache and DASD Fast Write Information 184
83. Displaying Storage Group Status for Each System/System Group 185
84. Displaying the SMS Volume Status 186
85. Displaying Status Information for Individual DASD Volumes 188
86. Displaying Status Information for Individual Optical Volumes 188
87. Displaying the Status of a Tape Volume 189
88. DEVSERV Command Syntax 189
89. Displaying the Status of a Device 190
90. Specifying Data Class View Criteria, Page 1 of 2 192
91. Specifying Data Class View Criteria, Page 2 of 2 192
92. Specifying Data Class Sort Criteria, Page 1 of 2 193
93. Specifying Data Class Sort Criteria, Page 2 of 2 193
94. Displaying a List of Data Classes 194
95. Moving a Volume from One Storage Group to Another 196
96. Altering a Management Class 197
97. Altering a Storage Class 199
98. Data Class Alter Panel, Page 1 of 3 200
99. Data Class Alter Panel, Page 2 of 3 200
100. Data Class Alter Panel, Page 3 of 3 201
101. Aggregate Group Alter, Page 1 of 2 202
102. Aggregate Group Alter, Page 2 of 2 203
103. Copy Entry Panel. 204
104. Confirm Delete Request Panel 208
105. Example of Global Connectivity in a Parallel Sysplex. 232
106. Example of Insufficient Connectivity in a Parallel Sysplex 232

107. JCL Example for Creating Sharing Control Data Sets 235
108. CDS (Control Data Set) Application Selection for VSAM RLS. 242
109. Coupling Facility Cache Set Definition for VSAM RLS 243
110. Example of CF Cache Structure Definition in Base Configuration 244
111. Defining a Storage Class for VSAM RLS 245
112. Defining Storage Class Attributes for VSAM RLS 245
113. Example of Constraints when Using the &ALLVOL and &ANYVOL Read-Only Variables. 275
114. Example Showing &JOB Read-Only Variable 276
115. Example Showing &PGM Read-Only Variable 276
116. Example Showing &USER Read-Only Variable 276
117. Example of a FILTLIST Statement 278
118. Using the INCLUDE and EXCLUDE 279
119. Example of a DO Statement. 280
120. Examples of IF Statements 281
121. Example of a SELECT Statement. 282
122. Example of an EXIT Statement. 283
123. Example of an END Statement. 284
124. Production ACS Routine for Storage Class 285
125. Entry Panel for the Data Collection Application, Page 1 of 2 302
126. Entry Panel for the Data Collection Application, Page 2 of 2 304
127. Entry Panel for the Data Collection Application, Page 3 of 3 305

Notices

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates.

Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of the intellectual property rights of IBM may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, programs, or services, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
USA

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Information Enabling Requests
Dept. DWZ
5600 Cottle Road
San Jose, CA 95193

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Any pointers in this publication to non-IBM Web sites are provided for convenience only, and do not in any manner serve as an endorsement of these Web sites.

Programming Interface Information

This book is intended to help you to perform storage management tasks. This book primarily documents General-use Programming Interface and Associated Guidance Information provided by DFSMS/MVS. General-use programming interfaces allow the customer to write programs that obtain the services of DFSMS/MVS.

This book also documents Diagnosis, Modification or Tuning Information which is provided to help you to customize DFSMS/MVS. Diagnosis, Modification or Tuning Information is identified where it occurs, either by an introductory statement to a chapter or section, or by the following marking:

NOT Programming Interface information

Diagnosis, Modification or Tuning Information

End of NOT Programming Interface information

Attention: Do not use this Diagnosis, Modification or Tuning Information as a programming interface.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States, or other countries, or both:

ADSTAR	IBM
Advanced Function Printing	IMS
Application System/400	IMS/ESA
AS/400	MVS/DFP
CICS	MVS/ESA
DATABASE 2	MVS/SP
DB2	MVS/XA
DFSMS	OpenEdition
DFSMS/MVS	OS/2
DFSMSdfp	OS/390
DFSMSdss	Parallel Sysplex
DFSMShsm	RACF
DFSMSrmm	RAMAC
DFSORT	RETAIN
Enterprise Systems Architecture/390	RMF
ESA/390	S/370
ESCON	S/390
Hiperbatch	System/390
Hiperspace	VTAM

UNIX is a registered trademark in the United States and/or other countries licensed exclusively through X/Open Company Limited.

Other company, product, and service names, which might be denoted by a double asterisk (**), might be trademarks or service marks of others.

About This Book

This book is intended for storage administrators who manage storage under DFSMS/MVS. This book explains how to initialize and maintain the DFSMS/MVS *Storage Management Subsystem* (SMS), and how to perform storage management tasks with the help of *Interactive Storage Management Facility* (ISMF) panels.

Required Product Knowledge

To use this book effectively, you should be familiar with storage management concepts. This book assumes you have begun preparing for system-managed storage by implementing some of the recommendations in *DFSMS/MVS Implementing System-Managed Storage*. You should also be familiar with the information presented in the following publications:

- *MVS/ESA SML: Managing Data*
- *MVS/ESA SML: Managing Storage Groups*

To get the most out of this manual, you should have access to the applications and functions available through the ISMF Primary Option Menu for Storage Administrators. If you want to use the applications and functions available through the ISMF Primary Option Menu for End Users, see *DFSMS/MVS Using ISMF*. For information on installing ISMF, see *DFSMS/MVS Planning for Installation*.

For information about using NaviQuest to perform batch functions such as testing ACS routines, define or alter pool or tape storage groups, and define or alter selected management class attributes, see *DFSMS/MVS NaviQuest User's Guide*.

You should also be familiar with:

- **Data Set Services (DFSMSdss)**. DFSMSdss moves data from one device to another, backs up and recovers data sets, and reduces free-space fragmentation on DASD volumes.

  For information about DFSMSdss, refer to the *DFSMS/MVS DFSMSdss Storage Administration Reference* .
- **Hierarchical Storage Manager (DFSMShsm)**. DFSMShsm provides automatic space management and availability functions through a hierarchy of storage devices.

  For information about DFSMShsm, see the *DFSMS/MVS DFSMShsm Storage Administration Guide*.
- **Device Support Facility (ICKDSF)**. ICKDSF initializes and formats DASD volumes, and recovers data from defective tracks.

  For information about ICKDSF, refer to the *ICKDSF R16 User's Guide*
- **Object Access Method (OAM)**. OAM manages objects on DASD or optical media. OAM also provides support for the IBM 3495 Automated Tape Library Dataserver.

  For information on objects and how to manage them, see *DFSMS/MVS OAM Application Programmer's Reference*, and *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support*.

For information on the IBM 3495 Tape Library Dataserver and OAM, see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Tape Libraries*.

For information on the IBM 3495 Tape Library Dataserver hardware, see *3495 Tape Library Dataserver Introduction* and *3495 Tape Library Dataserver Operator's Guide*

How to Tell if this Book is Current

IBM regularly updates its books with new and changed information. When first published, both hardcopy and BookManager softcopy versions of a book are identical, but subsequent updates might be available in softcopy before they are available in hardcopy. Here's how to determine the level of a book:

- Check the book's order number suffix (often referred to as the dash level). A book with a higher dash level is more current than one with a lower dash level. For example, in the publication order number SC26-4930-02, the dash level 02 means that the book is more current than previous levels, such as 01 or 00. Suffix numbers are updated as a product moves from release to release, as well as for hardcopy updates within a given release.
- Check to see if you are using the latest softcopy version. To do this, compare the last two characters of the book's file name (also called the book name). The higher the number, the more recent the book. For example, DGT1U302 is more recent than DGT1U301.
- Compare the dates of the hardcopy and softcopy versions of the books. Even if the hardcopy and softcopy versions of the book have the same dash level, the softcopy could be more current. This will not be apparent from looking at the edition notice. The edition notice number and date remain that of the last hardcopy version. When you are looking at the softcopy product bookshelf, check the date shown to the right of the book title. This will be the date that the softcopy version was created.

  Also, an asterisk (*) is added next to the new and changed book titles in the CD-ROM booklet and the README files.

Vertical lines to the left of the text indicate changes or additions to the text and illustrations. For a book that has been updated in softcopy only, the vertical lines indicate changes made since the last printed version.

Referenced Publications

Note: Some publications from the MVS/ESA System Product Version 5 library are referred to in this book. The *OS/390 Information Roadmap* contains a complete list of the MVS/ESA System Product Version 5 publications, and identifies any publications that are new or that replace publications in the previous version.

Within the text, references are made to the following publications:

Title	Order Number
IBM 3495 Tape Library Dataserver Introduction	GA32-0234
IBM 3495 Tape Library Dataserver Operator's Guide	GA32-0235
Character Data Representation Architecture Reference and Registry	SC09-2190
CICS Transaction Server for OS/390 Migration Guide	GC34-5353
Recovery and Restart Guide	SC33-1698
CICS Transaction Server for OS/390 Release Guide	GC34-5352

Title	Order Number
CICS Transaction Server for OS/390: Planning for InstallationCICS Transaction Server for OS/390: Planning for Installation	GC33-1789
DFSMS/MVS Access Method Services for ICF	SC26-4906
DFSMS/MVS DFSMSdss Storage Administration Guide	SC26-4930
DFSMS/MVS DFSMSdss Storage Administration Reference	SC26-4929
DFSMS/MVS DFSMSdfp Diagnosis Reference	LY27-9606
DFSMS/MVS DFSMShsm Storage Administration Guide	SH21-1076
DFSMS/MVS General Information	GC26-4900
DFSMS/MVS Installation Exits	SC26-4908
DFSMS/MVS Implementing System-Managed Storage	SC26-3123
DFSMS/MVS Managing Catalogs	SC26-4914
DFSMS/MVS Managing Data Availability	SC26-4928
DFSMS/MVS NaviQuest User's Guide	SC26-7194
DFSMS/MVS OAM Application Programmer's Reference	SC26-4917
DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support	SC26-4918
DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Tape Libraries	SC26-3051
DFSMS/MVS Planning for Installation	SC26-4919
DFSMS/MVS Using Data Sets	SC26-4922
DFSMS/MVS Using ISMF	SC26-4911
Device Support Facilities User's Guide and Reference	GC35-0033
International Technical Support Centers Implementing Concurrent Copy	GG24-3990
OS/390 ISPF Planning and Customizing	SC28-1298
OS/390 ISPF User's Guide	SC28-1239
MVS/ESA SML: Leading a Storage Administration Group	SC26-3126
MVS/ESA SML: Managing Data	SC26-3124
MVS/ESA SML: Managing Storage Groups	SC26-3125
OS/390 MVS Initialization and Tuning Guide	SC28-1751
OS/390 MVS JCL Reference	GC28-1757
OS/390 MVS JCL User's Guide	GC28-1758
OS/390 Information Roadmap	GC28-1727
OS/390 MVS System Management Facilities (SMF)	GC28-1783
OS/390 MVS System Commands	GC28-1781
MVS/ESA System Messages, Volumes 1–5	GC28-1480, GC28-1481, GC28-1482, GC28-1483, GC28-1484
OS/390 Security Server (RACF) Security Administrator's Guide	SC28-1915
Storage Subsystem Library Master Bibliography, Index, and Glossary	GC26-4496

Title	Order Number
OS/390 Parallel Sysplex Application Migration	GC28-1863
OS/390 Parallel Sysplex Hardware and Software Migration	GC28-1862
OS/390 TSO/E Customization	SC28-1965

References to Product Names Used in DFSMS/MVS Publications

DFSMS/MVS publications support DFSMS/MVS, 5695-DF1, as well as the DFSMSdfp base element and the DFSMShsm, DFSMSdss, and DFSMSrmm features of OS/390, 5647-A01. DFSMS/MVS publications also describe how DFSMS/MVS interacts with other IBM products to perform the essential data, storage, program and device management functions of the operating system.

DFSMS/MVS publications typically refer to another IBM product using a generic name for the product. When a particular release level of a product is relevant, the reference includes the complete name of that product. This section explains the naming conventions used in the DFSMS/MVS library for the following products:

MVS can refer to:

- MVS/ESA SP Version 5, 5695-047 or 5695-048
- The MVS base control program (BCP) of OS/390, 5647-A01

All MVS book titles used in DFSMS/MVS publications refer to the OS/390 editions. Users of MVS/ESA SP Version 5 should use the corresponding MVS/ESA book. Refer to *OS/390 Information Roadmap* for titles and order numbers for all the elements and features of OS/390.

For more information about OS/390 elements and features, including their relationship to MVS/ESA SP and related products, please refer to *OS/390 Planning for Installation*.

RACF can refer to:

- Resource Access Control Facility (RACF), Version 2, 5695-039
- The RACF element of the OS/390 Security Server, an optional feature of OS/390

All RACF book titles refer to the Security Server editions. Users of RACF Version 2 should use the corresponding book for their level of the product. Refer to *OS/390 Security Server (RACF) Introduction* for more information about the Security Server.

CICS can refer to:

- CICS/MVS, 5665-403
- CICS/ESA, 5685-083
- The CICS element of the CICS Transaction Server for OS/390, 5665-147

All CICS book titles refer to the CICS Transaction Server for OS/390 editions. Users of CICS/MVS and CICS/ESA should use the corresponding books for those products. Please see *CICS Transaction Server for OS/390: Planning for Installation* for more information.

Notational Conventions

A uniform notation describes the syntax of commands. This notation is not part of the language; it is merely a way of describing the syntax of the commands. The command syntax definitions in this book use the following conventions:

[ ] Brackets enclose an optional entry. You can, but need not, include the entry. Examples are:

[*length***]**

[MF=E]

| An OR sign (a vertical bar) separates alternative entries. You must specify one, and only one, of the entries unless you allow an indicated default. Examples are:

[REREAD|LEAVE]

[*length***|'S']**

{ } Braces enclose alternative entries. You must use one, and only one, of the entries. Examples are:

BFTEK={S|A}

{K|D}

{*address***|S|O}**

Sometimes alternative entries are shown in a vertical stack of braces. An example is:

MACRF={{(R[C|P])}{(W[C|P|L])}
{(R[C],W[C])}}

In the example above, you must choose only one entry from the vertical stack.

. . . An ellipsis indicates that the entry immediately preceding the ellipsis might be repeated. For example:

(*dcbaddr***,[(***options***)],. . .)**

‘ ’ A **‘ ’** indicates that a blank (an empty space) must be present before the next parameter.

UPPERCASE BOLDFACE
Uppercase boldface type indicates entries that you must code exactly as shown. These entries consist of keywords and the following punctuation symbols: commas, parentheses, and equal signs. Examples are:

CLOSE , , , ,TYPE=T

MACRF=(PL,PTC)

UNDERSCORED UPPERCASE BOLDFACE
Underscored uppercase boldface type indicates the default used if you do not specify any of the alternatives. Examples are:

[EROPT={ACC|SKP|ABE}]

[BFALN={F|D}]

Lowercase Italic
Lowercase italic type indicates a value to be supplied by you, the user, usually according to specifications and limits described for each parameter. Examples are:

number
image-id
count.

Summary of Changes

This section describes specific changes changes to this book for this and prior releases.

Sixth Edition, March 1999

This publication is a major revision in support of the functional changes introduced with DFSMS/MVS Version 1 Release 5. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

This revision also includes maintenance and editorial changes.

The following summarizes changes to that information:

- Changed occurrences of OS/390 OpenEdition to OS/390 UNIX System Services or its abbreviated name, OS/390 UNIX. As part of the name change of OpenEdition to OS/390 UNIX System Services, OpenEdition might continue to appear in messages, panel text, and other code with OS/390 UNIX System Services.
- Moved the section, “Running SMS in a Parallel Sysplex Environment” on page 5, from Chapter 3. Creating the Base Configuration to Chapter 1. Introducing the Storage Management Subsystem.
- Added a note explaining that the SMS control data set structure has changed dramatically with the functions introduced in DFSMS/MVS 1.5, so that DFSMS/MVS 1.5 systems in an SMS complex can only share SMS control data sets with DFSMS/MVS 1.2 and above systems, in “Running with Mixed Levels of DSMS/MVS” on page 7, in Chapter 1. Introducing the Storage Management Subsystem.
- Clarified the description of copying SCDSs and ACDSs, and added a recommendation that the REUSE option be used when defining an ACDS or SCDS, in “Calculating the Size of Storage and Active Control Data Sets” on page 11.
- Updated the description of object and object backup storage groups, in “Chapter 4. Defining Storage Groups” on page 37.
- Clarified the description of the partial release attribute and added a list of the conditions under which partial release is ignored, in “Partial Release Attribute” on page 73, in Chapter 5. Defining Management Classes.
- Added a description of the new Versioning and Backup subparameters supporting point-in-time copy using IBM RAMAC Virtual Array devices with SnapShot copy support and the DFSMSdss virtual concurrent copy SPE installed, in “Defining Accessibility” on page 93, in Chapter 6. Defining Storage Classes. This includes adding a new table, Table 4 on page 95, identifying the various ways in which you can request point-in-time copy devices.
- Modified Figure 41 on page 103 to accurately reflect volume selection based on accessibility decisions and requests for point-in-time copy volumes, in Chapter 6. Defining Storage Classes.
- Updated the description of data set type attributes to reflect that the extended format, extended addressability, and record access bias attributes are now available to all VSAM data set types, in “Defining Data Set Type Attributes for Data Class” on page 123, in Chapter 7. Defining Data Classes.

- Updated Figure 46 on page 120 to reflect that you can select the type of compression, using either tailored or generic dictionaries, for physical sequential data sets, in Chapter 7. Defining Data Classes.
- Added information to “Chapter 9. Defining ACS Routines” on page 139 on a new pre-ACS routine exit which a tape management system can use to set four new read-only variables, used by SMS for tape allocations. The four new variables are:
  – &MSDEST
  – &MSPOOL
  – &MSPOLICY
  – &MSPARM
- Changed the recommended logical record length to 80 or greater for the partitioned data set you need to allocate when you test ACS routines, in “Creating ACS Test Cases” on page 152, in Chapter 9. Defining ACS Routines.
- Clarified how processing occurs on a data set restore, recall or recover procedure and recommended that you not use the storage group ACS routine to test the value in &ACSENVIR, as this could yield inconsistent results, in “DFSMShsm” on page 166, in Chapter 9. Defining ACS Routines.
- Added a note explaining why using VOL=REF processing with temporary data sets might result in data sets being assigned to different storage groups, in Chapter 9. Defining ACS Routines.
- Revised the list of SMS parameters for the SETSMS operator command to delete several IGDSMSxx parameters in the list which cannot be set or changed by the SETSMS command, in “Changing Storage Management Subsystem Parameters” on page 173, in Chapter 10. Activating Storage Management Subsystem Configurations.
- Added a description and sample output of the D SMS,OAMXCF operator command, and updated OAM output examples, in “Chapter 11. Maintaining the Storage Management Subsystem” on page 179.
- Recommended that you copy SMS constructs from lists generated from an SCDS and not from an ACDS, in “Copying SMS Classes, Storage Groups, and Aggregate Groups” on page 204, in Chapter 11. Maintaining the Storage Management Subsystem.
- Added a description of four new read-only variables, in “Read-Only Variables” on page 261, in Chapter 15. ACS Language Reference.
- Added a list of variables that are returned on a data set rename using &ENVIR=RENAME, in “Using Read-Only Variables” on page 269, in Chapter 15. ACS Language Reference.
- Identified which read-only variables are not passed in different environments, in “Read-Only Variables in Different Environments” on page 271, in Chapter 15. ACS Language Reference.
- Updated the following ISMF panels:
  – ISMF Primary Option Menu
  – Object Storage Group Define
  – Storage Class Define (Page 1 of 2)
  – Storage Class Define (Page 2 of 2)
  – ACS Test Case Define (Page 4 of 4)
- Added the following new ISMF panels:

- ACS Test Case Define (Page 3 of 4)
- Data Collection Entry Panel (page 3 of 3)

Note: For other important updates to this book, please check informational APAR II11474, a repository of DFSMS/MVS 1.5 information that was not available at the time DFSMS/MVS books were published for general availability.

Fifth Edition, June 1997

This publication is a major revision in support of the functional changes introduced with DFSMS/MVS Version 1 Release 4. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

This revision also includes maintenance and editorial changes.

The following summarizes changes to that information.

- Added information about NaviQuest to "Using the Interactive Storage Management Facility (ISMF)" on page 2 in Chapter 1. Introducing the Storage Management Subsystem, "Chapter 9. Defining ACS Routines" on page 139, and "Chapter 11. Maintaining the Storage Management Subsystem" on page 179.
- Updated the sizing formula in "Calculating the Size of Storage and Active Control Data Sets" on page 11, in Chapter 2. Preparing for the Storage Management Subsystem.
- Added information about using compression specifically tailored to a data set and the new COMPRESS parameter for the IGDSMSxx parmlib member in "Modifying the SYS1.PARMLIB Data Set" on page 15, in Chapter 2. Preparing for the Storage Management Subsystem.
- Corrected the description of the management class Backup Copy Technique attribute, and added a caution on not specifying conflicting conditions in the storage class and the management class, in "Defining Management Class Attributes" on page 67, in Chapter 5. Defining Management Classes.
- Added MSR values for the following devices to Figure 34 on page 88 in Chapter 6. Defining Storage Classes: 9393, 9396, 9397.
- Updated the description on defining guaranteed space for VSAM key-sequenced data sets with key ranges specified in "Defining Storage Class Attributes" on page 85 in Chapter 6. Defining Storage Classes.
- Clarified the description of Figure 37 on page 97 in Chapter 6. Defining Storage Classes.
- Added more information on why a volume is not placed on the primary list, how striping volume selection occurs, and some possible reasons for volume selection failure in "Understanding Volume Selection" on page 99, in Chapter 6. Defining Storage Classes.
- Updated the description of data class parameters in "Defining Data Class Attributes" on page 116, in Chapter 7. Defining Data Classes, to include the following:
  - The new values for Media Type and Recording Technology
  - The new Record Access Bias attribute for specifying system-managed buffering
  - The new Spanned/Nonspanned attribute for specifying whether a data record can span control interval boundaries

– The new BWO attribute for specifying whether backup-while-open processing is to be used
– The new Log and Logstream Id attributes for specifying whether a data set is recoverable or not and providing the name of the forward recovery log stream.
– The new Space Constraint Relief and Reduce Space Up To % attributes for requesting to retry new data set allocations or extends on new volumes that fail due to space constraints, and specifying the percentage by which requested space is to be reduced when the allocation is retried.

- Clarified information on ACS testing in Figure 67 on page 155, in Chapter 9. Defining ACS Routines.
- Updated “Changing Storage Management Subsystem Parameters” on page 173 in Chapter 10. Activating Storage Management Subsystem Configurations to include all IGDSMSxx parameters.
- Added CLIST and module names for the ISMF QSAVE and QRETRIEV commands in Table 15 on page 219, in Chapter 13. Protecting the Storage Management Subsystem.
- Added information about the STGADMIN.IGG.DEFDEL.UALIAS RACF FACILITY profile which controls the ability to define or delete an alias related to a user catalog, in “Storage Administration (STGADMIN) Profiles in the FACILITY Class” on page 221, in Chapter 13. Protecting the Storage Management Subsystem.
- Updated the following information in “Chapter 14. Administering VSAM Record-Level Sharing” on page 229:
  – General rules and considerations to follow, in “Understanding the Product Environment for VSAM RLS” on page 230
  – Description of sharing control, in “Defining Sharing Control Data Sets” on page 233
  – Determining CF cache structure size, in “Defining CF Cache Structures” on page 235
  – Sizing formula for estimating the size of the CF lock structure, and the description of considerations for retained locks and record table full conditions, the discussion of false contention, and the lock structure sizing example, in “Defining the CF Lock Structure” on page 237
- Added information about read-only variables, specifically &RECORG, in “Read-Only Variables” on page 261, in Chapter 15. ACS Language Reference.
- Updated the following ISMF panels:
  – ISMF Primary Option Menu
  – VIO Storage Group Define (Page 1 of 2)
  – Data Class Define (Page 2 of 3)
  – Data Class Define (Page 3 of 3)
  – Aggregate Group Define (Page 1 of 2)
  – ACS Test Case Define (Page 1 of 3)
  – Data Class Alter (Page 2 of 3)
  – Data Class Alter (Page 3 of 3)
  – Aggregate Group Alter (Page 1 of 2)
  – Data Class View Entry (Page 1 of 2)
  – Data Class View Entry (Page 2 of 2)
  – Data Class Sort Entry (Page 1 of 2)

– Data Class Sort Entry (Page 2 of 2)

Fourth Edition, September 1996

This publication is a revision in support of the functional changes introduced with DFSMS/MVS Version 1 Release 3. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

This revision also includes maintenance and editorial changes.

Following are the changes that are specific to this book:

- Added information about the following new parameters for the IGDSMSxx parmlib member in "Modifying the SYS1.PARMLIB Data Set" on page 15, in Chapter 2. Preparing for the Storage Management Subsystem:
  - HSP_SIZE
  - RLSINIT
  - RLS_MAX_POOL_SIZE

  The descriptions of the CF_TIME and SMF_TIME parameters in that same section were also updated.
- Added information, including examples, on using keyword parameters to code an IEFSSNxx member in "Modifying the SYS1.PARMLIB Data Set" on page 15, in Chapter 2. Preparing for the Storage Management Subsystem.
- Added warnings against defining an SMS complex that spans a Parallel Sysplex, as well as defining duplicate SMS configurations that use the same DASD, in "Specifying Systems and System Groups in the SMS Complex" on page 31, in Chapter 3. Creating the Base Configuration.
- Updated the description of weight attributes for data that is eligible for VSAM record-level sharing, in "Defining Use of the Coupling Facility for VSAM Record-Level Sharing" on page 97, in Chapter 6. Defining Storage Classes.
- Added a note indicating that special considerations might be needed when moving control data sets and their catalog entries to "Notes on Recovering Control Data Sets" on page 210, in Chapter 12. Recovering Storage Management Subsystem Information.
- Completely reworked and updated the information in "Chapter 14. Administering VSAM Record-Level Sharing" on page 229, as follows:
  - Added a new section, "Understanding the Product Environment for VSAM RLS" on page 230, that describes some general rules to consider when falling back from or moving forward to CICS, CICSVR or DFSMS/MVS release levels.
  - Added information about defining sharing control data sets (SHCDSs) to "Defining Sharing Control Data Sets" on page 233, including details about what sharing options to use.
  - Updated "Defining CF Cache Structures" on page 235 with new information on determining coupling facility (CF) cache structure sizes, including a description on how to limit the maximum size of local buffer pools using the new RLS_MAX_POOL_SIZE parameter.
  - Updated "Defining the CF Lock Structure" on page 237 with a new, easier-to-use sizing formula, along with new information on retained locks,

how to avoid false contention, and how to adjust the size of the lock structure. We also provided a lock structure sizing example.

– Updated “Modifying the SYS1.PARMLIB IGDSMSxx Member” on page 241 with the new IGDSMSxx parmlib parameters, RLSINIT and RLS_MAX_POOL_SIZE.

– Reworked information in “Establishing Authorization for VSAM RLS” on page 241 .

– Modified Figure 110 on page 244.

– Updated the description of weight attributes for data that is eligible for VSAM record-level sharing, in “Defining Storage Classes for VSAM RLS” on page 244.

– Added information on when VSAM RLS processing is *not* available to “Activating VSAM RLS” on page 246, as well as details about how the LOG parameter must be specified on the DEFINE CLUSTER or ALTER CLUSTER command for the data set if that data set is to be opened for VSAM RLS processing.

– Reworked the information on falling back from VSAM RLS processing and created a new section, “Falling Back from VSAM RLS Processing” on page 254, that fully describes the rules and considerations, as well as the specific procedure to follow.

– Modified information in “Monitoring the Coupling Facility for VSAM RLS” on page 247 and “Recovering VSAM RLS Processing” on page 253.

- Corrected the description of the &DSNTYPE read-only variable in “Read-Only Variables” on page 261 and the description of boolean expressions in “Boolean Expressions” on page 273, in Chapter 15. ACS Language Reference.

Chapter 1. Introducing the Storage Management Subsystem

The *Storage Management Subsystem* (SMS), introduced by MVS/DFP 3.1, provides a range of data and space management functions. SMS improves storage space use, controls external storage centrally, and lets you manage storage growth. It makes it easier to convert to new device types. It takes advantage of what available hardware can do. With SMS, you can move toward system-managed storage.

This chapter introduces using SMS to define your storage management policy, and using the *Interactive Storage Management Facility (ISMF)* to define and manage your SMS configurations. It also outlines the steps for activating an SMS configuration, and discusses some considerations for running SMS in a multiprocessor environment.

Defining a Storage Management Policy

SMS manages an installation's storage according to the currently active storage management policy. Through ISMF, you define an installation storage management policy in an *SMS configuration*. An SMS configuration contains the following:

- Base configuration information
- Classes and groups
- Automatic class selection (ACS) routines
- Optical library and drive definitions
- Tape library definitions

The *base configuration* identifies the systems that the SMS configuration manages. These systems constitute an *SMS complex*. The base configuration also contains installation defaults.

You can define more than one control data set, but only one at a time controls SMS. Each control data set defined for SMS is called a *source control data set (SCDS)*. The control data set that is in effect at a given time is the *active control data set (ACDS).*

SMS classes and groups are lists of traits and characteristics that are associated with or assigned to data sets, objects and volumes. An SMS configuration can contain the following five types of classes and groups:

Storage group
: allows you to define a list of volumes and manage them as if they were one large, single volume. SMS applies the properties you assign to a storage group to all the volumes within the storage group.

Management class
: allows you to define different levels of migration, backup and retention services. Through management class, you can associate a level of service with a data set or object that is independent of the physical location of the data set or object. Also, you can identify an object characteristic that might trigger a class transition.

Storage class
: allows you to define different levels of performance and availability services. Through storage class, you can separate the level of service for a data set or object from physical device characteristics. You can also separate the

level of service for an object with different storage classes used to place objects at various levels of the storage hierarchy.

Data class
allows you to define allocation defaults. Through data class, you can simplify and standardize the allocation of new data sets.

Aggregate Group
allows you to define groups of data sets for the purpose of backing up or recovering all data sets in a group in a single operation.

An SMS configuration can contain multiple constructs of each type. Data sets managed by SMS are called *system-managed*. Each system-managed data set or object must reside in a storage group. The system-managed data sets must have a storage class, and might also have a management class and a data class. The objects must have a storage class, a management class, and cannot have a data class.

You can assign the same name to various SMS classes and a storage group. For example, a data class and a storage class can have the same name.

ACS routines determine the SMS classes and storage groups for data sets and objects. You can also use ACS routines to control the transition of data sets and objects to and from SMS management.

Using the Interactive Storage Management Facility (ISMF)

ISMF provides a series of applications for storage administrators to define and manage SMS configurations. You can use these applications to:

- Define SMS base configuration information.
- Define, alter, delete, or copy individual SMS classes, storage groups, aggregate groups, optical libraries, optical drives, and tape libraries.
- Display parameters and values of individual SMS classes, storage groups, aggregate groups, mountable optical volumes, optical drives, mountable tape volumes and tape libraries.
- Generate, save and manage lists of SMS classes, storage groups, aggregate groups, mountable optical volumes, optical libraries, optical drives, mountable tape volumes and tape libraries.
- Edit ACS routines.
- Define, alter, and execute ACS test cases.
- Validate the correctness and completeness of an SMS configuration.
- Activate an SMS configuration.
- Display, define, alter, or delete storage group information pertaining to specific volumes using AUDIT, EJECT, ALTER, and RECOVER (RECOVER is for optical volumes only).
- Produce data set, volume, or capacity planning measurement data.
- Maintain mountable optical volumes and mountable tape volumes.
- Use DFSMSrmm to maintain tape volumes.
- Use the DFSMS/MVS NaviQuest tool to perform enhanced testing of your ACS routines, and to perform many storage management tasks in batch, such as:
  – Updating and testing your base configuration
  – Translating and testing your ACS routines

– Generating test cases from previously collected DCOLLECT data
– Defining, altering, and displaying information for management classes, data classes and storage classes
– Defining or altering information for storage groups
– Defining, altering, and displaying information for the base configuration, as well as for aggregate groups
– Generating data set and volume lists and reports
– Diagnosing data set and volume problems

For more information on the DFSMS/MVS NaviQuest tool, see the *NaviQuest User's Guide*.

Figure 1 shows the ISMF Primary Option Menu for storage administrators.

```
  Panel  Help
------------------------------------------------------------------------------
                   ISMF PRIMARY OPTION MENU - DFSMS/MVS 1.5
Enter Selection or Command ===>

Select one of the following options and press Enter:

0  ISMF Profile             - Specify ISMF User Profile
1  Data Set                 - Perform Functions Against Data Sets
2  Volume                   - Perform Functions Against Volumes
3  Management Class         - Specify Data Set Backup and Migration Criteria
4  Data Class               - Specify Data Set Allocation Parameters
5  Storage Class            - Specify Data Set Performance and Availability
6  Storage Group            - Specify Volume Names and Free Space Thresholds
7  Automatic Class Selection - Specify ACS Routines and Test Criteria
8  Control Data Set         - Specify System Names and Default Criteria
9  Aggregate Group          - Specify Data Set Recovery Parameters
10 Library Management       - Specify Library and Drive Configurations
11 Enhanced ACS Management  - Perform Enhanced Test/Configuration Management
C  Data Collection          - Process Data Collection Function
L  List                     - Perform Functions Against Saved ISMF Lists
R  Removable Media Manager  - Perform Functions Against Removable Media
X  Exit                     - Terminate ISMF
Use HELP Command for Help; Use END Command or X to Exit.
```

Figure 1. ISMF Primary Option Menu for Storage Administrators

This primary option menu differs from the one that end users see. To the options found on the ISMF Primary Option Menu for End Users, this primary option menu adds the following:

- Storage Group
- Automatic Class Selection
- Control Data Set
- Library Management
- Data Collection

Additionally, the management class, data class, storage class, and aggregate group applications available through the ISMF Primary Option Menu for Storage Administrators allow you to define, alter, copy, and delete SMS classes. End users can list the available SMS classes, display the attributes of individual SMS classes, and list or display volumes and datasets.

Implementing the Storage Management Subsystem

Before you implement a storage management policy by activating an SMS configuration, you need to follow the steps outlined in Figure 2. The chapters in this manual that describe these steps appear to the right of the respective boxes.

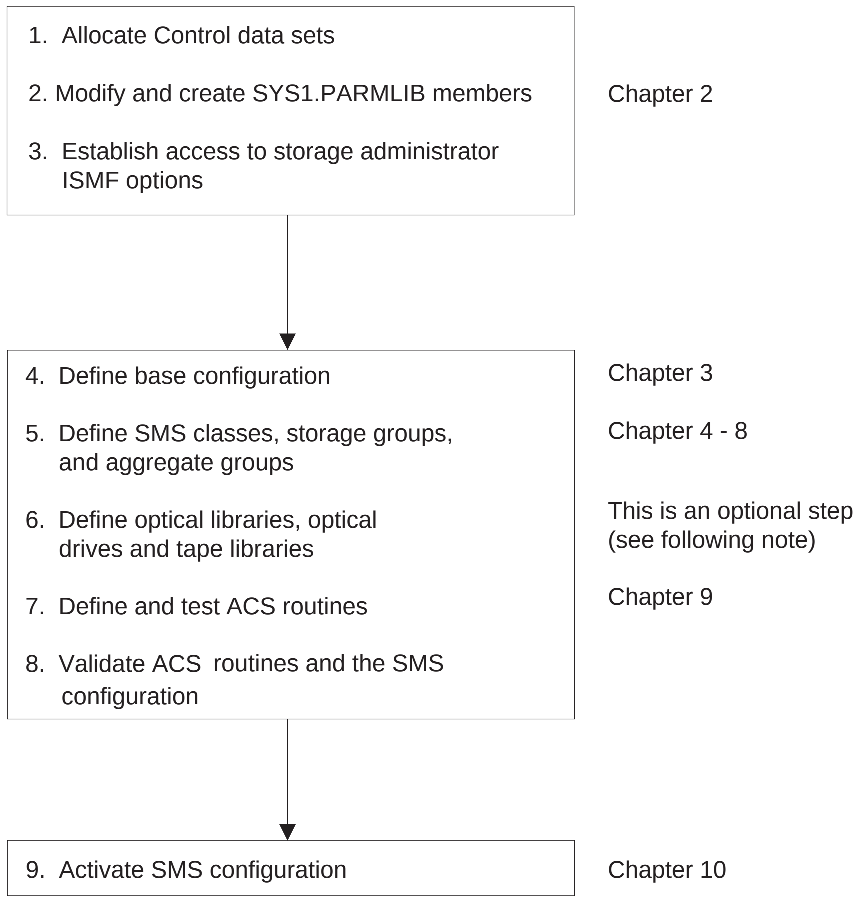

Figure 2. Activating an SMS Configuration

Note: See *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support* and *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Tape Libraries* for information on step 6.

When preparing for SMS, you need to:

1. Allocate *control data sets*, which contains information used by SMS.
2. Modify and create SYS1.PARMLIB members to identify SMS to all the systems in the SMS complex. These new members take effect when you IPL.
3. Establish access to the ISMF Primary Option Menu for Storage Administrators, which is shown in Figure 1 on page 3. This is the final preparation step.
4. Define the base configuration, which identifies the systems within the SMS complex.
5. Define the SMS classes and storage groups that you want SMS to assign to your data sets and objects, and the data sets that you want to assign to your aggregate groups.

6. Define optical libraries and drives and tape libraries, if you have them.
7. Define ACS routines to assign the SMS classes and storage groups. Then you test the routines.
8. Validate the ACS routines individually to check for errors. You should then validate the entire SMS configuration to check for errors that exist among its related parts.
9. Activate the valid SMS configuration.

Running SMS in a Parallel Sysplex Environment

An SMS complex consists of systems or *system groups* that share a common configuration. A Parallel Sysplex is made up of systems that share a cross-system coupling facility (XCF); you can run multiple SMS complexes within a Parallel Sysplex.

Notes:

1. We strongly recommend that an SMS complex not span sysplexes. All of the volumes in the SMS complex should be in the same Parallel Sysplex, since cross-system sharing functions, such as VSAM record-level sharing (RLS), PDSE sharing, RACF security and global shared resources (GRS) serialization only work within the scope of a single Parallel Sysplex. These types of functions are not supported when the SMS complex extends beyond the Parallel Sysplex in which they are carried out.
2. We also recommend that you do *not* set up multiple SMS complexes sharing the same DASD. This not only requires extra work to maintain the duplicate SMS configurations but can also create problems such as running out of disk space, since one configuration cannot know about changes made to the other configuration, such as data set allocations and deletions, and storage group and volume status changes.

Basic Terms and Definitions

Following are some basic terms and definitions:

SMS complex
: A system or a collection of systems that share a common configuration including a common active control data set (ACDS) and a common communication data set (COMMDS) pair. The SMS configuration now supports up to 32 system names, system group names, or both.

Parallel Sysplex
: A collection of MVS systems in a multi-system environment supported by the cross-system coupling facility (XCF).

System name
: The name of the system where an SMS operation is being performed.

System group
: The system names within a Parallel Sysplex, excluding those systems in that Parallel Sysplex, if any, that are individually defined in the SCDS. This support can be used even if the XCF is not active.

System Grouping Concept

In the MVS environment, a Parallel Sysplex is a collection of systems linked by closely coupled hardware facilities to process customer workloads. SMS system group name support allows you to specify a system group as a member of an SMS complex.

In SMS, the term *system group* is used instead of Parallel Sysplex. A system group consists of system names within a Parallel Sysplex, excluding those systems that are individually specified in the SMS base configuration. A system group name can represent multiple systems. This allows SMS to support more than eight systems per SMS complex while retaining the existing configuration format of the configuration data set.

A system group name matches a Parallel Sysplex name and refers to all systems defined as part of the Parallel Sysplex that are:

- Running the same SMS configuration
- Defined in the configuration using the name of the Parallel Sysplex to which they belong (that is, system group)
- NOT defined in the configuration by their system names.

The system group name represents all the systems in the Parallel Sysplex which are not explicitly specified in the SCDS base configuration.

Figure 3 on page 7 provides a visual representation of the system grouping concept.

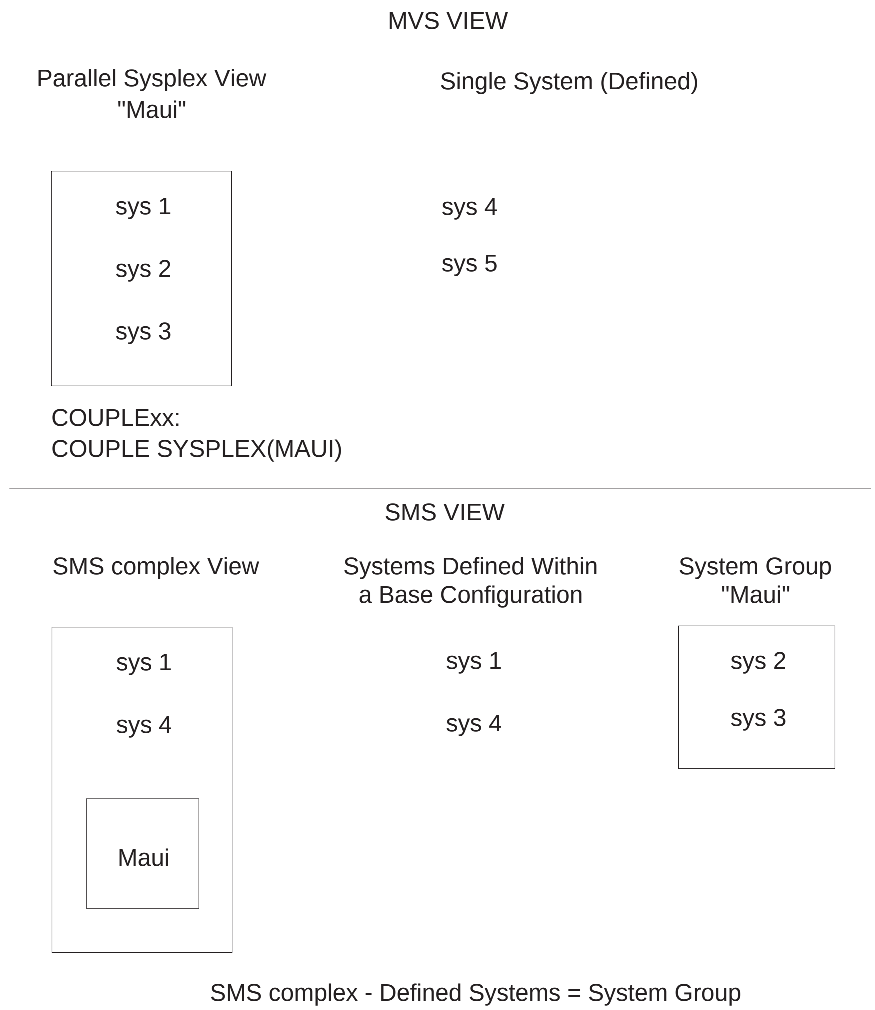

Figure 3. System Grouping Concept

In Figure 3, the SMS complex view shows that it is managing sys 1 and sys 4 as single systems, and Maui as a Parallel Sysplex. (sys 5 is not part of the SMS complex.)

The Parallel Sysplex (Maui) contains three system names: sys 1, sys 2, and sys 3. The single systems sys 1 and sys 4, as well as the Parallel Sysplex (Maui) are all defined in the SCDS. If you perform an SMS operation on Maui, it affects only sys 2 and sys 3 in the Parallel Sysplex. It does not affect sys 1 because it is defined separately in the SCDS.

In this example, the system group (sys 2 and sys 3) is represented by the Parallel Sysplex name, Maui.

Running with Mixed Levels of DSMS/MVS

When running in an SMS complex with mixed levels of DFSMS/MVS you must consider compatibility across the complex. See “Translating and Validating in a Multiprocessor Environment” on page 150 and “Calculating the Size of Storage and Active Control Data Sets” on page 11 for details.

Also be aware that the structure of SMS control data sets changed dramatically with the functions introduced in DFSMS/MVS 1.3 and DFSMS/MVS 1.5. Maintenance is available for DFSMS/MVS 1.2, and above, to allow DFSMS/MVS 1.5 to share SMS control data sets with lower level systems in the SMS complex. MVS/DFP V3 and DFSMS/MVS 1.1 systems cannot share SMS control data sets with systems at the DFSMS/MVS 1.5 level; however, they can share DASD data sets and volumes that are not SMS-managed.

In addition, sharing control data sets with DFSMS/MVS 1.2 is possible only if you run DFSMS/MVS in *compatibility mode*. With compatibility mode, no more than eight system names can be defined in the SMS configuration. This enables the system to share the SMS control data sets with DFSMS/MVS 1.2, and with other systems running in compatibility mode.

Chapter 2. Preparing for the Storage Management Subsystem

Before you define and activate an SMS configuration, you need to perform the following preparatory steps:

- Allocate control data sets to contain your SMS configuration and to permit the systems in your complex to communicate with each other.
- Modify SYS1.PARMLIB, which contains three members that direct the initialization and activation of SMS.
- Establish access to the ISMF Primary Option Menu for Storage Administrators (shown in Figure 1 on page 3).

This chapter describes how to perform these preliminary steps so that you can begin defining a base configuration for an SMS configuration.

For additional information regarding the planning and preparation for implementing SMS, see *DFSMS/MVS Implementing System-Managed Storage*.

When planning and preparing for SMS use with optical and tape libraries see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support* and *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Tape Libraries*.

Allocating Control Data Sets

| **Note:** You should be aware that the structure of SMS control data sets changed dramatically with the functions introduced in DFSMS/MVS 1.3 and DFSMS/MVS 1.5. Maintenance is available for DFSMS/MVS 1.2, and above, to allow DFSMS/MVS 1.5 to share SMS control data sets with lower level systems in the SMS complex. MVS/DFP V3 and DFSMS/MVS 1.1 systems cannot share SMS control data sets with systems at the DFSMS/MVS 1.5 level; however, they can share DASD data sets and volumes that are not SMS-managed.

| In addition, sharing control data sets with DFSMS/MVS 1.2 is possible only if you run DFSMS/MVS in *compatibility mode*. With compatibility mode, no more than eight system names can be defined in the SMS configuration. This enables the system to share the SMS control data sets with DFSMS/MVS 1.2, and with other systems running in compatibility mode.

| Before you can activate an SMS configuration, you need to allocate control data sets used by SMS, and define their contents. Control data sets are virtual storage access method (VSAM) linear data sets that contain base configuration information, SMS class, aggregate group, optical library, tape library, optical drive, and storage group definitions, and ACS routines.

Before the SMS address space is activated, you can allocate control data sets with access method services or TSO/E commands (explained in this section). You define and alter the contents of control data sets using ISMF. SMS uses three types of control data sets: a *source control data set* (SCDS), an *active control data set* (ACDS), and a *communications data set* (COMMDS).

Note: Do not name any of your SMS control data sets ’ACTIVE’. SMS uses the single word ’ACTIVE’ as a reserved word indicating the active configuration residing in the SMS address space. Naming an SMS control data set ’ACTIVE’ results in errors.

In order to integrate DFSMS/MVS 1.5 with previous releases of DFP or DFSMS, migrating systems need to share configurations and communications data sets (COMMDs) with both the newer and older releases. To facilitate this, SMS supports a compatibility mode in which systems starting with DFSMS/MVS 1.3 support only eight systems in its configuration. It is important to note that DFSMS/MVS 1.3 or above systems must run in compatibility mode if they share SMS control data sets with systems running earlier releases of DFSMS/MVS or MVS/DFP. This allows you to migrate your systems to DFSMS/MVS 1.5 one at a time rather than having to convert them all at once. This is described in detail with the systems parameter of “Initializing SMS through the IGDSMSxx Member” on page 15.

Source Control Data Set (SCDS)

An SCDS contains an SMS configuration, which defines a storage management policy. You can define any number of SMS configurations each of which has its own SCDS. Then, you select one SMS configuration to be the installation storage management policy and make an active working copy of it in an ACDS.

Active Control Data Set (ACDS)

When you activate an SCDS, its contents are copied to an ACDS. The current ACDS contains a copy of the most recently activated configuration. All systems in an SMS complex use this configuration to manage storage. You can define any number of SCDSs, but only one can be put in the ACDS. “Initializing SMS through the IGDSMSxx Member” on page 15 explains how to specify the ACDS. You can define more than one IGDSMSxx member, each specifying a different ACDS, but you can use only one ACDS at a time.

Note: You cannot define or alter an ACDS. This also means that you cannot use an ACDS as an SCDS if the SCDS is lost.

You can modify the SCDS from which your current storage management policy was activated without disrupting operations, because SMS manages storage with a copy of the SMS configuration (an ACDS) rather than with the original (an SCDS). While SMS manages storage using an ACDS, you can:

- Create a backup copy of the SCDS
- Build a new SCDS
- Update the SCDS from which the ACDS was activated
- Modify any SCDS

Figure 4 on page 11 shows the relationship among SCDSs and ACDSs in an installation.

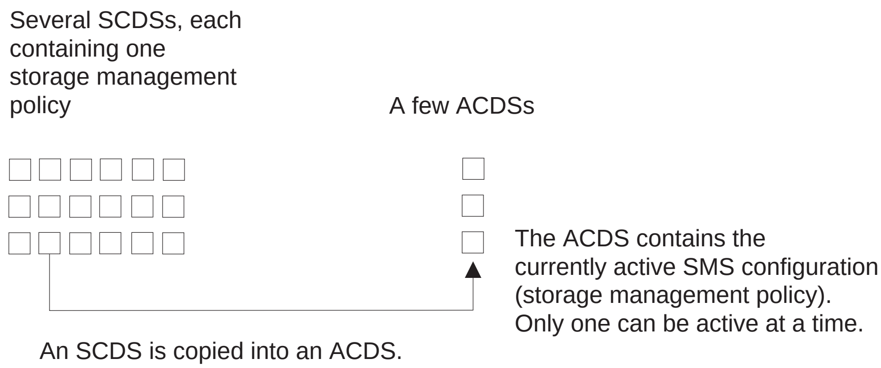

Figure 4. Relationship among SCDSs and ACDSs in an Installation

Communications Data Set (COMMDS)

The COMMDS serves as the primary means of SMS communication among systems in the SMS complex. The active systems in an SMS complex access the COMMDS for current SMS complex information.

The COMMDS contains the name of the ACDS containing the currently active storage management policy, the current utilization statistics for each system-managed volume, and other system information. You can define any number of COMMDSs, but only one can be active in an SMS complex.

Calculating the Size of Storage and Active Control Data Sets

Before you allocate control data sets, you need to estimate their size. When calculating the size of either an ACDS or an SCDS, you have to account for system and base configuration information, SMS class, aggregate group, storage group, optical library and drive, and tape library definitions.

With SMS 32-name support, the amount of space required for an SCDS or ACDS has increased. SMS 32-name support allows up to 32 unique system names, system group names, or both to be specified in the configuration. Previously allocated SCDSs and ACDSs might have insufficient space to support the configuration changes. You might need to allocate new SCDSs and ACDSs and copy your old SCDSs into new SCDSs and your old ACDSs into new ACDSs prior to activating SMS on a DFSMS/MVS 1.5 system.

The following formula can help you determine the size of a source or active control data set:

```
size (bytes) = 150000 +
               (14000 * SG) +
               (2312 * (MC + SC + DC + AG + CS)) +
               (12000 * (DRV + LIB + VOL))
```

where:

150000 bytes
represent fixed data fields. For example, if you have

Five storage groups

Four management classes

Two storage classes

Five data classes

Two aggregate groups
6 optical libraries
24 optical drives
40 DASD volumes.

the equation yields a value of 1088600 bytes for control data set allocation.

SG is the estimated number of storage groups

MC is the estimated number of management classes

SC is the estimated number of storage classes

DC is the estimated number of data classes

AG is the estimated number of aggregate groups

CS is the estimated number of cache sets in the base configuration (for record-level sharing (RLS) only)

DRV is the estimated number of optical drives in the SMS complex to be managed by SMS

LIB is the estimated number of optical and tape libraries in the SMS complex to be managed by SMS

VOL is the estimated number of DASD volumes in the SMS complex to be managed by SMS

If you are running SMS in a multiprocessor environment with different releases of MVS/DFP and DFSMS/MVS, you must allocate your control data sets using calculations that are based on the highest level of DFSMS/MVS. For example, if you are running with MVS/DFP 3.3 and DFSMS/MVS, you must use the calculations given for DFSMS/MVS. If you do not, your control data sets might be too small, because the storage requirements for classes and groups might be different between releases or new classes, groups, or other items might be added.

| If your SCDS or ACDS is not large enough, you might receive SMS reason code 6068 when attempting to save its contents. When reason code 6068 occurs, allocate a new, larger control data set and copy your existing SCDS into your new SCDS, or your existing ACDS into your new ACDS. You can then delete the old SCDS or ACDS and use the new one.

Allocating a new SCDS or ACDS resolves the problem only when reason code 6068 is caused by a data set size problem. Because this reason code is returned when a system service called by data-in-virtual (DIV) fails, the error might have other causes. Messages returned to you or the system console can help determine the cause of the failure.

Calculating the Size of a COMMDS

When you calculate the size of a COMMDS, you have to account for both system and volume information. With SMS 32-name support, the amount of space required for a COMMDS has increased. A previously allocated COMMDS might have insufficient space to support the changes. You might need to allocate a new COMMDS prior to activating SMS on a DFSMS/MVS 1.5 system. The following formula helps you determine the size of a COMMDS: (VOL = Estimated number of DASD volumes in the SMS complex to be managed by SMS)

COMMDS size (bytes) =
8192 + (588 * VOL)

As an example, if you have 40 DASD volumes in the SMS complex, you need to allocate 31712 bytes for the COMMDS.

Selecting Volumes for Control Data Sets

SMS control data sets can be either SMS-managed or non-SMS-managed. Initially you should ensure that your control data sets have a volume count of one. The volume count can be either explicitly specified, implied by the number of volume serials provided, or derived from the data class assigned to the data set (see “Chapter 7. Defining Data Classes” on page 115 for more information). If you give an SMS control data set a volume count that is greater than the number of volumes on which it actually resides you might receive messages IEF244I and IEF489I when you attempt to activate it.

If a control data set grows to a size where it needs to become a multivolume data set, you can add volumes using the access method services ALTER ADDVOLUMES command. For more information on this command see *DFSMS/MVS Access Method Services for ICF*.

If you have a multivolume SCDS which you are activating into a single volume ACDS, you might receive an error because the ACDS is not large enough and volumes cannot be dynamically added to it. To bypass this problem, you need to create a new multivolume ACDS and then activate the ACDS and SCDS simultaneously using the SETSMS command (for more information on this command, see “Changing Storage Management Subsystem Parameters” on page 173).

| If your SMS complex includes more than 16 systems, be sure that the ACDS and COMMDS are accessible to every system in the complex. Define your control data sets on volumes which are capable of being attached to more than 16 systems, such as IBM RAMAC Virtual Array volumes.

Allocating an SCDS

The following access method services job allocates a 6-track SCDS.

```
//STEP  EXEC PGM=IDCAMS
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  DEFINE CLUSTER(NAME(SMS.SCDS1.SCDS) LINEAR VOL(SMSV01) -
  TRK(6 6) SHAREOPTIONS(2,3)) -
  DATA(NAME(SMS.SCDS1.SCDS.DATA))
/*
```

This job creates a VSAM linear data set named SMS.SCDS1.SCDS. You can combine the DEFINE commands for all your allocations into the same job step, but this example shows only one for purposes of illustration. After allocating an SCDS, you define its contents through ISMF dialogs.

You should allocate an SCDS on a device shared by all systems in the SMS complex. If you allocate an SCDS on a device that is not shared by all the systems, then you can activate the SCDS only from systems that have access to it.

You should specify the REUSE option when you define an SCDS to avoid running into space problems (SMS reason code 6068) as result of subsequent SCDS updates, or IMPORT/EXPORT functions.

For information on using access method services commands, see *DFSMS/MVS Access Method Services for ICF*.

Allocating an ACDS

The following access method services job allocates a 6-track ACDS.

```
//STEP  EXEC PGM=IDCAMS
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  DEFINE CLUSTER(NAME(SMS.ACDS1.ACDS) LINEAR VOL(SMSV02) -
  TRK(6 6) SHAREOPTIONS(3,3)) -
  DATA(NAME(SMS.ACDS1.ACDS.DATA))
/*
```

This job creates a VSAM linear data set named SMS.ACDS1.ACDS.

An ACDS must reside on a shared volume, accessible from all systems in the SMS complex. To ease recovery in case of failure, the ACDS should reside on a different volume than the COMMDS. Also, you should allocate a spare ACDS on a different shared volume. “Chapter 12. Recovering Storage Management Subsystem Information” on page 209 provides additional information on the backup and recovery of control data sets.

You create the contents of an ACDS by activating a valid SCDS. The distinction between a valid SCDS and one that is not valid is described in “Defining the Base Configuration” on page 27. The control data set (ACDS or COMMDS) must reside on a volume that is not reserved by other systems for a long period of time because the control data set (ACDS or COMMDS) must be available to access for SMS processing to continue.

You should specify the REUSE option when you define an ACDS to avoid running into space problems (SMS reason code 6068) as result of subsequent ACDS updates, or IMPORT/EXPORT functions.

Allocating a COMMDS

The following access method services job allocates a 1-track COMMDS;

```
//STEP  EXEC PGM=IDCAMS
//SYSUDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSIN    DD *
  DEFINE CLUSTER(NAME(SMS.COMMDS1.COMMDS) LINEAR VOL(SMSVOL) -
  TRK(1 1) SHAREOPTIONS(3,3)) -
  DATA(NAME(SMS.COMMDS1.COMMDS.DATA))
/*
```

This job creates a VSAM linear data set named SMS.COMMDS1.COMMDS.

The COMMDS must reside on a shared volume accessible from all systems in the SMS complex. To ease recovery in case of failure, the COMMDS should reside on a different volume than the ACDS. Also, you should allocate a spare COMMDS on a different shared volume. “Chapter 12. Recovering Storage Management

Subsystem Information" on page 209 provides additional information on the backup and recovery of control data sets. The control data set (ACDS or COMMDS) must reside on a volume that is not reserved by other systems for a long period of time because the control data set (ACDS or COMMDS) must be available to access for SMS processing to continue.

Specifying Share Options for an ACDS and COMMDS

We recommend that you use SHAREOPTIONS(3,3) when allocating an ACDS. This allows full authority to read from and write to an ACDS from any system. The ACDS and COMMDS must be accessed from all systems in the complex simultaneously.

Modifying the SYS1.PARMLIB Data Set

The IGDSMS*xx*, IEASYS*yy*, and IEFSSNxx members of SYS1.PARMLIB direct the initialization and activation of SMS. IGDSMS*xx* provides initialization parameters to SMS. The SMS=*xx* parameter of IEASYS*yy* indicates the name of the SYS1.PARMLIB member IGDSMS*xx* that is used for initialization. For example, if SMS=01 in IEASYS*yy*, then the IGDSMS01 member of SYS1.PARMLIB is used during initialization. The SMS entry in IEFSSNxx identifies SMS to MVS.

This section explains how to create an IGDSMS*xx* member in SYS1.PARMLIB and describes two methods to define SMS to MVS through IEFSSNxx. For additional information, see *OS/390 MVS Initialization and Tuning Guide*.

Initializing SMS through the IGDSMSxx Member

For each system in the SMS complex, you must create an IGDSMS*xx* member in SYS1.PARMLIB. The IGDSMS*xx* member contains SMS initialization control information, where *xx* is any value allowed by the naming conventions for SYS1.PARMLIB members. It has a default value of 00.

Every SMS system must have an IGDSMS*xx* member in SYS1.PARMLIB that specifies a required ACDS and COMMDS pair. This ACDS and COMMDS pair is used if the COMMDS of the pair does not point to another COMMDS.

If the COMMDS points to another COMMDS, the referenced COMMDS is used. This referenced COMMDS might contain the name of an ACDS that is different from the one specified in the SYS1.PARMLIB member. If so, the name of the ACDS is obtained from the COMMDS rather than from the IGDSMS*xx* member in SYS1.PARMLIB to ensure that the system is always running under the most recent ACDS and COMMDS.

If the COMMDS of the ACDS and COMMDS pair refers to another COMMDS during IPL, it means a more recent COMMDS has been used. SMS uses the most recent COMMDS to ensure that you cannot IPL with a down-level configuration.

The data sets that you specify for the ACDS and COMMDS pair must be the same for every system in an SMS complex. Whenever you change the ACDS or COMMDS, update the IGDSMS*xx* member of SYS1.PARMLIB for every system in the SMS complex so that it specifies the same data sets.

The IGDSMS*xx* member contains the following parameters, which direct the SMS initialization process and specify the names of the ACDS and COMMDS:

SMS	**ACDS(***dsname***) COMMDS(***dsname***)[ACSDEFAULTS({YES|NO})]** **[ASID({asid|*})]** **[BMFTIME({nnn|3600})]** **[CACHETIME({nnn|3600})]** **[CF_TIME({nnn|3600})]** **[COMPRESS({TAILORED|GENERIC})]** **[DB2SSID(***ssid***)]** **[DEADLOCK_DETECTION({iiii|15,kkkk|4})]** **[DESELECT({event[,event][,...]|ALL})]** **[DINTERVAL({nnn|150})]** **[DSNTYPE({LIBRARY|PDS})]** **[HSP_SIZE(***nnn***)]** **[INTERVAL({nnn|15})]** **[JOBNAME({jobname|*})]** **[OAMTASK(***taskid***)]** **[OVRD_EXPDT({YES|NO})]** **[PDSESHARING({NORMAL|EXTENDED})]** **[REVERIFY({YES|NO})]** **[RLSINIT({NO|YES})]** **[RLS_MAX_POOL_SIZE({nnnn|100})]** **[SELECT({event[,event][...]|ALL})]** **[SIZE(***nnn***{K|M})]** **[SMF_TIME({YES|NO})]** **[SYSTEMS({32|8})]** **[TRACE({OFF|ON})]** **[TRACEEXIT(***user_trace_exit***)]** **[TYPE({ALL|ERROR})]** **[USE_RESOWNER({YES|NO})]**

SMS, ACDS, COMMDS, REVERIFY, and ACSDEFAULTS need to be the same for all your systems. The rest of the parameters can be different. For more information on the parameters of the IGDSMS*xx* member of SYS1.PARMLIB, see *OS/390 MVS Initialization and Tuning Guide*.

Required Keywords

Blanks are permitted between SMS keywords.

SMS Identifies the member as a repository of SMS initialization control information.

ACDS(*dsname***)**
identifies the name of the data set containing the active configuration. If you omit *dsname*, the operator is prompted for a value.

COMMDS(*dsname***)**
identifies the name of the COMMDS. If you omit *dsname*, the operator is prompted for a value.

Optional Keywords

ACSDEFAULTS({YES|NO})
indicates whether SMS initializes the following ACS routine variables from an additional call to RACF:

&APPLIC

&DEF_DATACLAS

&DEF_MGMTCLAS

&DEF_STORCLAS

If you specify **NO**, these variables have no values associated with them. The default value for ACSDEFAULTS is **NO**. The **ACSDEFAULTS** keyword is not applicable for OAM.

ASID({*asid*|*})
: limits tracing to a certain address space or permits it for all address spaces. Specify * if you want all address spaces traced (providing SMS tracing is activate). This is the default. You can enter up to 4 digits for the ASID keyword. If you leave off the leading zeroes, they are inserted.

BMFTIME({*nnn*|3600})
: specifies the number of seconds between recording SMF record type 42, subtype 1 records for PDSE and HFS I/O (where the buffer hits are I/O requests which did not result in actual I/O being done). You can specify a value from 1 to 86399 (23 hours, 59 minutes, 59 seconds), and the default is 3600 (one hour).

CACHETIME({*nnn*|3600})
: specifies the number of seconds between recording SMF records for device cache use. The **CACHETIME** parameter applies only to the volumes behind an IBM 3990 Storage Control with cache unit. You can specify a value from 1 to 86399 (23 hours, 59 minutes, 59 seconds), and the default is 3600 (one hour).

CF_TIME({*nnn*|3600})
: indicates the number of seconds between recording SMF records for the coupling facility (both cache and lock). You can specify a value from 1 to 86399 (23 hours, 59 minutes, 59 seconds), and the default is 3600 (one hour). This keyword sets the interval time for the following SMF 42 subtypes:

SUBTYPE 15
: Coupling facility storage class average response time

SUBTYPE 16
: Coupling facility data set average response time

SUBTYPE 17
: Coupling facility lock structure activity

SUBTYPE 18
: Coupling facility cache partition summary

SUBTYPE 19
: SMSVSAM least recently used statistics summary

[COMPRESS({TAILORED|GENERIC})]
: Specifies the type of compression to be used for the data set.

TAILORED
: specifies that the data set is eligible for compression specifically tailored to the data set. A tailored dictionary is built, using the initial data written to the data set, and imbedded into the data set. The dictionary is used to compress or expand data written to or read from the data set. This type of compression applies only to sequential data sets, not to VSAM KSDSs.

To convert an existing DBB-based compressed data set to use tailored compression, you must set the COMPRESS parameter to

TAILORED and copy the generic DBB-based data set to a new data set that meets compression requirements.

GENERIC
specifies that the data set be compressed using generic Dictionary Building Block (DBB) compression. The dictionary is derived from a defined set of compression algorithms in data set SYS1.DBBLIB. This is the default.

Notes:

1. Use tailored compression only when all systems in the SMS complex have been converted to DFSMS/MVS 1.4 or higher, and when there is no need to revert to a prior release level for local recovery or remote recovery with Aggregate Backup and Recovery Support (ABARS).
2. To convert an existing DBB-based compressed data set to use tailored compression, first set the COMPRESS parameter to TAILORED in the IGDSMSxx parmlib member. Use IEBGENER, ICEGENER, REPRO, or any QSAM or BSAM application to copy the DBB-based data set to a new data set that meets compression requirements.

DB2SSID(*ssid*)
| identifies either the name of the DB2 subsystem used by OAM for object storage, or a group attachment name if data sharing is being used. If you specify a group attachment name, the DB2 startup ECB is ignored by DB2, so DB2 must be initialized before OAM is started.

| If OAM object support and DB2 are not needed for OAM's object directories, object data tables, and configuration database, this parameter is not necessary and should not be specified. The *ssid* can be from one to four characters and there is no default.

[DEADLOCK_DETECTION({iiii|15,kkkk|4})]
specifies the deadlock detection intervals used by the DFSMSdfp Storage Management Locking Services. The first subparameter is the local deadlock detection cycle and specifies the interval in seconds for detecting deadlocks within a system. The second subparameter is the global deadlock detection cycle and specifies the interval for detecting deadlocks between systems. The value is specified as the number of local detection cycles that occur before global deadlock detection is initiated.

iiii one to four digit numeric value in the range 1-9999 that specifies the length in seconds of the local deadlock detection interval. The default is 15 seconds.

kkkk one to four digit numeric value in the range 1-9999 that specifies the number of local deadlock cycles that must complete before global deadlock detection occurs. The default is 4.

DESELECT({event[,event][,...]|ALL})
deletes items from the list of events to be traced (if SMS tracing is active). DESELECT has no default. If you specify events that conflict in SELECT and DESELECT, the keyword that appears last has final authority.

The events that you can specify on SELECT and DESELECT are:

MODULE
SMS module entry or exit

DSTACK
SMS Data Set Stacking Service

SMSSJF
SMS/SJF interfaces

SMSSSI
SMS/SSI interfaces

ACSINT
ACS services interfaces

OPCMD
Operator commands

CONFC
Configuration change

CDSC Control data set changes

CONFS
SMS Configuration services

MSG SMS Message services

ERR SMS Error recovery and recording services

CONFR
Return data from an active configuration

CONFA
Activate a new configuration

ACSPRO
Perform ACS processing

IDAX SMS interpreter/dynamic allocation

DISP SMS disposition processing exit

CATG SMS catalog services

VOLREF
SMS VOLREF services

SCHEDP
SMS scheduling services (preallocate catalog orientation)

SCHEDS
SMS scheduling services (system select)

VTOCL
SMS VTOC/data set services (allocate existing data set)

VTOCD
SMS VTOC/data set services (delete existing data set)

VTOCR
SMS VTOC/data set services (rename existing data set)

VTOCC
SMS VTOC/data set services (allocate new data set)

VTOCA
SMS VTOC/data set services (add a volume to a data set)

RCD SMS Recording services

DCF SMS device control facility

DPN SMS device pool name select subsystem interface

TVR SMS tape volume record update facility

ALL All of the above options

DINTERVAL({*nnn*|150})
specifies the number of seconds SMS allows to elapse before it reads device statistics. The **DINTERVAL** parameter applies only to the volumes behind an IBM 3990 Storage Control with cache unit. You can specify a value from 1 to 999 (16 minutes, 39 seconds), and the default is 150.

Note: SMS uses device statistics to manage hardware usage and maximize efficiency.

DSNTYPE({LIBRARY|PDS})
specifies the installation default for data sets allocated with directory space but without a data set type specified. If the data set type is **PDS**, the default is a partitioned data set format; if the data set type is **LIBRARY**, the default is a PDSE (partitioned data set extended) format.

[HSP_SIZE(*nnn*)]
specifies the size, in megabytes, of the hiperspace that is used to cache PDSE member pages. You can specify a hiperspace size up to 512MB. The valid range for HSP_SIZE is 0 to 512. If you specify 0, the hiperspace is not created. To create a hiperspace, your system must have expanded storage available. If expanded storage is available and you do not specify a value for HSP_SIZE, the default is 256MB or half of expanded storage, whichever is less.

INTERVAL({*nnn*|15})
specifies the *synchronization time interval* of the system, which is the number of seconds between system checks of the COMMDS for news about SMS configuration changes from other systems in the SMS complex. You can specify a value from 1 to 999 (16 minutes, 39 seconds), and the default is 15.

JOBNAME({*jobname*|*})
| limits tracing to a certain job or permits tracing on all jobs. The default is to
| trace all jobs, *****.

OAMPROC(*procname*)
identifies the procedure name that starts the OAM address space when SMS is initialized. You must specify this keyword if you want the OAM address space started during IPL. The procedure name can be from one to eight characters and there is no default.

OAMTASK(*taskname*)
specifies the task ID that is used to start the OAM address space. **OAMTASK** is optional; if it is specified without **OAMPROC** it is ignored. If omitted, the task ID defaults to the procedure name specified in **OAMPROC**. OAM keywords take effect only if you initialize SMS at IPL; otherwise they are not saved. The task ID can be from one to eight characters.

OVRD_EXPDT({YES|NO})
allows for the override of an expiration date when an unexpired SMS-managed DASD data set is deleted. If you specify NO, expiration dates are honored unless specific action is taken to override them. If you specify YES, any expiration date specified for the data set is overridden when deletion of an SMS-managed DASD data set is attempted through JCL, SVC 99, IEHPROGM or the SCRATCH macro. YES should only be

used when management class cannot be used to override expiration dates. It is intended to help with data sets which are converted from tape to DASD since expiration date is more commonly used with tape.

PDSESHARING({NORMAL|EXTENDED})
specifies whether PDSEs can be shared by all systems within a Parallel Sysplex for input only or for input and output. A value of **NORMAL** means that PDSEs can be shared by all the systems in a Parallel Sysplex but *only* for input. A value of **EXTENDED** means that PDSEs can be shared by all the systems in a Parallel Sysplex for input *and* output. The default is **NORMAL**.

REVERIFY({YES|NO})
specifies whether SMS verifies authorization only at job interpretation time or at both job interpretation and job execution time. A value of **NO** instructs SMS to verify a user's authority to allocate a new data set, to use a storage class, and to use a management class at job interpretation time only. A value of **YES** instructs SMS to perform the verification at both job interpretation and job execution time. JES2 environments ignore the **REVERIFY** keyword. It has a default value of **NO**.

[RLSINIT({NO|YES})]
specify YES if you want the SMSVSAM address space started as part of system initialization or the V SMS,SMSVSAM,ACTIVE command. This value applies only to the system accessed by the parmlib member and is acted upon when SMSVSAM is next started. The default is NO.

[RLS_MAX_POOL_SIZE({nnnn|100})]
specifies the maximum size in megabytes of the SMSVSAM local buffer pool. SMSVSAM attempts to not exceed the buffer pool size you specify, although more storage might be temporarily used. Because SMSVSAM manages buffer pool space dynamically, this value does not set a static size for the buffer pool.

Use SMF 42, subtype 19 records to help you determine the maximum size of the SMSVSAM local buffer pool.

You can specify a two to four-digit numeric value, with 10 as the minimum value. If you specify a value less than 10, the field is set to 10. If you specify a value greater than 1500, SMSVSAM assumes there is no maximum limit. We recommend that you limit the size of the local buffer pool.

The default is **100MB**.

SELECT({*event*[, *event*][,...]|ALL})
adds items to the list of events to be traced (if SMS tracing is active). The default is **ALL**. See **DESELECT** for a list of valid events.

SIZE(*nnn*{K|M})
specifies the size of the SMS trace table in bytes. When the unit is **K**, the value can range from 0K to 255000K, and it is rounded up to the nearest 4K unit. When the unit is **M**, the value can range from 0M to 255M. If you specify a value of 0, no tracing is performed. The default is 128K.

[SMF_TIME({YES|NO})]
YES indicates that the following SMF type 42 records are created at the SMF interval time, and that all of the indicated records are synchronized with SMF and RMF data intervals:

SUBTYPE 1
Buffer management statistics

SUBTYPE 2
Cache control unit statistics (IBM 3990 Storage Control Model 3)

SUBTYPE 15
Coupling facility storage class average response time

SUBTYPE 16
Coupling facility data set average response time

SUBTYPE 17
Coupling facility lock structure activity

SUBTYPE 18
Coupling facility cache partition summary

SUBTYPE 19
SMSVSAM least recently used statistics summary

DFSMS creates the specified SMF record at the end of the interval period and SMF sends the event notification signal. If YES is specified, this subparameter overrides the following subparameters: BMFTIME, CACHETIME, CF_TIME. YES is the default.

See *OS/390 MVS System Management Facilities (SMF)* for more information on SMF.

SYSTEMS({32|8})
indicates whether the system is running in compatibility mode (eight name limit) or 32-name mode. The default is **8**.

SYSTEMS(8) specifies that a maximum of eight system names, system group names, or both, can be specified in the SMS configuration. This indicates that the system is running in compatibility mode and can share configurations (SCDSs or ACDSs) and COMMDSs with systems that are running down level releases of DFSMS/MVS or DFP. Essentially, the system continues to operate as it has in the past.

SYSTEMS(32) specifies that a maximum of 32 system names, system group names or both can be specified in the SMS configuration. This indicates the system is not running in compatibility mode.

Note: Therefore the ACDS, SCDS and COMMDS might not be shared with the systems that are running down level releases of DFSMS/MVS or DFP or that are running in compatibility mode.

TRACE({OFF|ON})
indicates whether the SMS trace facility is activated. The default is **ON**. The SMS trace facility records trace records in the SMS address space. See *DFSMS/MVS DFSMSdfp Diagnosis Reference* for additional information.

TRACEEXIT(*user_trace_exit***)**
specifies the name of your trace exit routine. You can specify an alphanumeric value of 1 to 8 characters, starting with an alphabetic character. Your value must represent a valid module name. The default value is blank. This interface is usually used only for diagnostic purposes under the guidance of the service organization and therefore is not documented here.

TYPE({ALL|ERROR})
specifies whether SMS traces all events or only errors. The default is **ERROR**.

USE_RESOWNER({YES|NO})
indicates whether construct authorization checking is done using the RESOWNER value, which is based on the high-level qualifier of the data set name, or using the data set allocator user ID. If you specify **NO**, the RESOWNER value is not extracted and the allocator user ID is used. The default value for USE_RESOWNER is **YES**.

Defining SMS to MVS through IEFSSNxx

Before you can activate SMS, MVS must recognize SMS as a valid subsystem. You can define SMS to MVS in one of two ways. The first method involves placing a record for SMS in the existing IEFSSNxx member that is used for the next initial program load (IPL). The format for the SMS entry (record) is shown in “Writing the SMS Record”.

The second method involves creating a new IEFSSNxx member that contains only the record for SMS. To identify the IEFSSNxx member containing the SMS entry, you must update the IEASYS*yy* member that you use to IPL. In the SSN parameter of IEASYS*yy*, the suffix identifying the IEFSSNxx member that contains the SMS entry should appear first, because SMS must be active before starting any other subsystem. For example, in:

```
SSN=(00,01,...)
```

the `00` identifies IEFSSN00 as the first member, and `01` identifies IEFSSN01 as the second member, and so forth. You can specify additional members to define more subsystems. For example:

```
SSN=(00,01,02,03).
```

You have the option of manually starting SMS with a command or automatically starting it at future IPLs. When you are first preparing for SMS, you might want to omit the IGDSSIIN module name from the SMS entry. This defines SMS to MVS at the next IPL but does not automatically start SMS. After you IPL the system, SMS is defined as a subsystem and you can then start it manually.

Writing the SMS Record

To define SMS to MVS, you must place a record for SMS in an IEFSSNxx member. You can code an IEFSSNxx member with keyword or positional parameters, but not both. We recommend using keyword parameters. Note also that you must use the same format for all systems on which you are running SMS.

The keyword syntax you can use to define SMS in IEFSSNxx is:

```
SUBSYS SUBNAME(SMS)
[INITRN(IGDSSIIN) [INITPARM('ID=yy,PROMPT=NO)]]
                                          YES)]]
                                          DISPLAY)]]
```

Figure 5. Keyword Syntax to Define SMS in IEFSSNxx

If you choose to use positional instead of keyword parameters, the positional format of the SMS definition in IEFSSNxx is:

```
SMS[,[IGDSSIIN][,'ID=yy[,PROMPT=NO]]
                               YES]]
                               DISPLAY]]
```

Figure 6. Positional Syntax to Define SMS in IEFSSNxx

Required Keywords:

SMS identifies and defines the Storage Management Subsystem to MVS.

Optional Keywords:

IGDSSIIN
identifies the subsystem initialization routine IGDSSIIN for SMS. If you include it in the SMS entry, SMS is automatically started during every future IPL. If you omit it from the SMS entry, SMS is defined as a valid subsystem to MVS, but does not automatically start during future IPLs.

ID={*xx*|00}
specifies the SYS1.PARMLIB member used for initialization in two special cases:

- When the value specified in the SMS parameter of IEASYS*yy* does not correspond to any IGDSMS*xx* and the default IGDSMS00 does not exist.
- When initialization of software controlling PDSEs fails.

To use PDSEs, SMS and the PDSE address space must be active. SMS allows you to allocate PDSEs and the PDSE address space allows you to open, read, write, and update PDSEs. If the PDSE address space initializes, the IGDSMS*xx* member used for SMS initialization is determined by the SMS=xx parameter in IEASYS*yy*. If the PDSE address space fails, the IGDSMS*xx* member used for SMS initialization is determined by the ID=xx parameter in IEFSSNxx. The SMS parameters should be the same unless you want different parameters for SMS when PDSEs are not available.

You should specify the same value in SMS=*xx* in IEASYS*yy* and in ID=*xx* in IEFSSNxx to avoid confusion.

If you specify both **ID** and **PROMPT**, enclose them in single quotation marks and separate them with a comma.

PROMPT={DISPLAY|YES|NO}
indicates how much control you want the operator to have during the rest of SMS initialization. DISPLAY displays the contents of the IGDSMS*xx* member, but the operator cannot modify them. YES displays the contents of the IGDSMS*xx* member and prompts the operator to modify the parameters in the IGDSMS*xx* member. The modifications take effect for the current IPL, but the contents of the IGDSMS*xx* SYS1.PARMLIB member do not change. NO, which is the default, neither displays the parameters nor allows the operator to modify them.

If you specify more than one keyword, enclose them in single quotation marks and separate them with a comma.

Starting the SMS Address Space

When you have completed preparations and are ready to start SMS, use the T SMS=*xx* command, where *xx* identifies IGDSMS*xx* as the SMS initialization member. The T SMS=*xx* command is an abbreviation for the SET SMS=*xx* command, discussed in "Step Two: Prepare One System" on page 171. To eliminate confusion with the SETSMS operator command, the abbreviated T SMS=*xx* form of the SET SMS=*xx* command is used throughout the remainder of this manual.

When you have sufficiently tested your operations and are ready to have SMS automatically started at future IPLs, add the IGDSSIIN module name to the SMS entry.

Examples of the SMS Record

- The following SMS record defines SMS to MVS without starting SMS at IPLs:
  ```
  SUBSYS SUBNAME(SMS)
  ```
- The following SMS record defines SMS to MVS and starts SMS at future IPLs:
  ```
  SUBSYS SUBNAME(SMS) INITRTN(IGDSSIIN)
  ```
  The system uses the default values of **ID** and **PROMPT**. IGDSMS00 specifies initialization information, and the operator has no control over the rest of SMS initialization.
- The following SMS record allows the operator to modify SMS initialization:
  ```
  SUBSYS SUBNAME(SMS) INITRTN(IGDSSIIN)
         INITPARM(',PROMPT=YES')
  ```
  The system uses the default value of **ID**, which identifies IGDSMS00 as containing initialization information. The **PROMPT** parameter requests that SMS display IGDSMS00, so that the operator can modify the parameters in IGDSMS00.
- The following SMS record initializes SMS using IGDSMS01:
  ```
  SUBSYS SUBNAME(SMS) INITRTN(IGDSSIIN)
         INITPARM('ID=01,PROMPT=DISPLAY')
  ```
  The **PROMPT** parameter requests that the contents of IGDSMS01 be displayed, but the operator cannot modify them.

Accessing the Storage Administrator Primary Option Menu

The first time you select ISMF, you get the ISMF Primary Option Menu for *end users*. To get the ISMF Primary Option Menu for *storage administrators* (which is shown in Figure 1 on page 3), select option 0, `ISMF PROFILE`. Within the ISMF Profile Option Menu, select option 0, `USER MODE`, and press ENTER. You get the User Mode Entry panel, where you indicate that you want the storage administrator Primary Option Menu for all future ISMF sessions. To do this, select option 2 on the User Mode Entry panel. After changing the user mode, you must exit ISMF and then return to it to view the Primary Option Menu for Storage Administrators.

"Chapter 13. Protecting the Storage Management Subsystem" on page 213 explains how to prevent end users from gaining access to the storage administrator Primary Option Menu through the ISMF PROFILE option. The reason for restricting access to the Primary Option Menu for Storage Administrators is to prevent unauthorized users from performing storage administrator tasks.

Planning for VSAM Record-Level Sharing

Planning for and installing VSAM record-level sharing requires coordination with system hardware and software groups. The following planning tasks are described fully in “Chapter 14. Administering VSAM Record-Level Sharing” on page 229:

- “Determining Hardware Requirements” on page 230
- “Determining Which Applications Can Use VSAM RLS” on page 231
- “Ensuring Same Systems Connectivity” on page 232
- “Planning for Availability” on page 233
- “Defining Sharing Control Data Sets” on page 233
- “Defining CF Cache Structures” on page 235
- “Defining the CF Lock Structure” on page 237
- “Modifying the SYS1.PARMLIB IGDSMSxx Member” on page 241
- “Establishing Authorization for VSAM RLS” on page 241

Chapter 3. Creating the Base Configuration

The first thing that you define in an SCDS is the base configuration. A base configuration contains installation defaults, such as a default management class, and identifies the systems to which the SMS configuration applies.

This chapter describes base configurations and explains how to define their contents using the ISMF control data set application.

Planning the Base Configuration

You need to determine the system names and system group names that you want to specify in the SCDS. These systems and system groups constitute the SMS complex.

Before defining the base configuration, consider what you want to do with system-managed data sets that do not have a management class. You can specify a default management class for these data sets in the base configuration. You can create the management class any time before validating the SCDS. Defining the management classes is described in “Chapter 5. Defining Management Classes” on page 65.

If the management class ACS routine does not determine a management class for a data set, DFSMShsm processes the data set using the default management class, if one exists. If you do not specify a default management class in your base configuration, DFSMShsm uses its own defaults.

If the management class ACS routine does not determine a management class for an *object*, the OSREQ request fails. This request can be OSREQ STORE, CHANGE, or OSMC class transition processing.

See *DFSMS/MVS Implementing System-Managed Storage* for additional planning information.

Defining the Base Configuration

After allocating an SCDS, you can define a base configuration. By selecting option 8 from the ISMF Primary Option Menu for Storage Administrators, you can invoke the Control Data Set (CDS) Application Selection panel shown in Figure 7 on page 28.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                          CDS APPLICATION SELECTION
Command ===>

To Perform Control Data Set Operations, Specify:
  CDS NAME . . 'ACTIVE'
                            (1 to 44 Character Data Set Name or 'Active')

Select one of the following Options:

 1   1. Display       - Display the Base Configuration
     2. Define        - Define the Base Configuration
     3. Alter         - Alter the Base Configuration
     4. Validate      - Validate the SCDS
     5. Activate      - Activate the CDS
     6. Cache Display - Display CF Cache Structure Names for all CF Cache Sets
     7. Cache Update  - Define/Alter/Delete CF Cache Sets

If CACHE Display is chosen, Enter CF Cache Set Name . . *
                             (1 to 8 character CF cache set name or * for all)

Use ENTER to Perform Selection;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 7. CDS Application Selection

On this panel, you can specify the name of the SCDS that is to contain the base configuration. ISMF primes the CDS NAME field with the last used SCDS name.

After specifying an SCDS, select option 1, Display, and press ENTER to see the base configuration. You can use this option to look at the base configuration of the current active SMS configuration by specifying 'ACTIVE' in the CDS NAME field.

If you want to define a base configuration, select option 2, Define, and press ENTER, to see the SCDS Base Define panel shown in Figure 8 on page 29.

If you want to alter an SCDS base configuration that you defined previously, select option 3, Alter, from the CDS Application Selection panel and press ENTER. The Alter panel contains the same fields as the Define panel.

To validate the entire SCDS or any of the ACS routines, select option 4, Validate, and press ENTER. To activate the CDS, select option 5, Activate, and press ENTER. The CDS NAME must be a valid source control data set.

Notes:

1. You must activate the configuration for any changes made to the SCDS to take effect.
2. The SMS configuration now supports up to 32 system names, system group names, or both. When the system is running in compatibility mode (8-name mode) you can specify a maximum of eight system names, system group names or both in the SMS configuration. When the system is running in 32-name mode, you can specify a maximum of 32 system, system group names or both.

   The current system mode is defined in the IGDSMSxx member of SYS1.PARMLIB, specified when SMS is started (see “Chapter 2. Preparing for the Storage Management Subsystem” on page 9). If SMS is not active, the number of systems you can put in the SCDS is determined by the SCDS itself.

If it is an 8-name configuration, you can only put eight names in it. If it is a 32-name configuration, you can put up to 32 names in it. If it is an empty data set, it defaults to 8-name mode.

3. When you access an SMS control data set (SCDS, ACDS, or COMMDS) for update which supports only eight names on a system running in 32-name mode, the data set must be converted to a new, incompatible format in order to support 32 names. You must confirm this conversion, either through the operator console or ISMF. This conversion is permanent, so you should make copies of your control data sets before converting the system mode from compatibility mode to 32-name mode.

If at any time you want to leave either the SCDS Base Define or Alter panels without saving the changed base configuration information, issue the CANCEL command.

Figure 8 shows page 1 of the SCDS Base Define panel.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                              SCDS BASE DEFINE                 Page 1 of 2
Command ===>

SCDS NAME . : SMS.SCDS1.SCDS
SCDS Status : INVALID

To DEFINE SCDS Base, Specify:

 Description ===> SYSTEM1 CONFIGURATION
             ===>

 Default Management Class  . . TEST      (1 to 8 characters)
 Default Unit  . . . . . . . . 3380      (esoteric or generic device name)
 Default Device Geometry
   Bytes/Track . . . . . . . . 47476     (1-999999)
   Tracks/Cylinder . . . . . . 15        (1-999999)



Use ENTER to Perform Verification; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL To Exit.
```

Figure 8. SCDS Base Define panel

Figure 9 on page 30 shows page 2 of the SCDS Base Define panel.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                              SCDS BASE DEFINE                 Page 2 of 2
Command ===>

SCDS NAME . : SMS.SCDS1.SCDS
SCDS Status : INVALID

Specify one of the following options . .           (1 Add, 2 Delete, 3 Rename)

  Specify System Name . . . . .           or Sys Group Name . .

  New System/Sys Group Name . .           (For option 3, Rename)

System: SYSTEM1  SYSTEM2  SYSTEM3  SYSTEM4  SYSTEM5  SYSTEM6  SYSTEM7  SYSTEM8



Sysgrp: SYSPLX01 SYSPLX02



Use ENTER to Perform Option; Use UP Command to View previous Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL To Exit.
```

Figure 9. SCDS Base Define panel

As its name implies, an SCDS base configuration forms the foundation for building an SCDS.

Before you can activate an SCDS, it should be validated. The SCDS STATUS field tells you whether the SCDS is valid or not valid. SMS sets the status when you save an SCDS with the END command or when you select the validate option from the CDS Application Selection panel.

Figure 61 on page 149, lists conditions that cause an SCDS to be not valid. "Validating ACS Routines or an Entire SCDS" on page 146 provides more information about the validate option.

You can use the Description field to help you identify and describe the SCDS. Your description can be up to 120 characters.

The following sections explain the content of the base configuration.

Specifying the Default Management Class

For system-managed data sets that have not been assigned a management class, DFSMShsm uses the default management class for expiration, migration, and backup information. The default management class does not apply to *objects*. You specify the name of the management class in the Default Management Class field.

The default management class name is not saved in the data set catalog entry, so that you can tell the difference between data sets with assigned management classes and those without them. Specifying a default management class is optional. If a data set has no management classes and a default management class is not defined, DFSMShsm uses its own defaults for the data set.

Specifying the Default Unit

The default unit is an esoteric or generic device name, such as SYSDA or 3390, that applies to data sets that are not managed by SMS. For new data set allocations, the default unit allows users to omit the UNIT parameter from DD statements or dynamic allocation equivalent, as they can when allocating system-managed data sets. The default unit does not apply to objects.

If users do not specify the unit parameter on their DD statements or the dynamic allocation equivalent, SMS applies the default unit if the data set is non-system managed and has a disposition of either MOD (treated as NEW) or NEW. If you specify a default unit in the base configuration, make certain that it exists on the system performing the allocations. If you do not specify a default unit, end users must code the UNIT parameter to allocate data sets that are not system-managed.

Specifying the Default Device Geometry

When allocating space for a new data set on DASD, SMS converts all space requests in tracks (TRK) or cylinders (CYL) into requests for space in KB or MB. If a generic device type such as the 3380 is specified, SMS uses the device geometry for that generic device to convert tracks or cylinders into KB or MB. If an esoteric device type such as SYSDA or no UNIT is specified, SMS uses the default device geometry to convert tracks and cylinders into KB or MB. If the users in your installations specify space in tracks or cylinder units, and they specify an esoteric UNIT or no UNIT, you must specify a default device geometry prior to converting these allocations to system-managed data sets.

After SMS converts space requests to KB or MB, the space values are passed to the ACS routines. The values are later used to determine the number of tracks or cylinders to allocate for the data set. The default device geometry does not apply to objects or data sets allocated on tape.

There is only one default device geometry for the entire SMS complex. Default device geometry is an installation's definition of how much space is represented by a TRK or a CYL when an esoteric unit or no unit is specified. The device *geometry* is the track size and number of tracks per cylinder for the device.

The device geometry for a 3380 is 47476 bytes/track, 15 tracks/cylinder. The device geometry for a 3390 is 56664 bytes/track, 15 tracks/cylinder. It is up to each installation to decide what values to use.

Bytes/Track represents the number of bytes per track that SMS uses on allocations.

Tracks/Cylinder represents the number of tracks per cylinder that SMS uses on allocations.

| Specifying Systems and System Groups in the SMS Complex

| You can add, delete, or rename the systems or system groups defined to an SMS
| complex using page 2 of the SCDS Base Define panel, shown in Figure 9 on
| page 30.

| A Parallel Sysplex is a collection of systems linked by closely coupled hardware
| facilities to process workloads. In SMS, the term *system group* is used instead of
| Parallel Sysplex. SMS system group name support allows you to specify a system

group as a member of an SMS complex. A system group consists of system names within a Parallel Sysplex, excluding those systems that are individually specified in the SMS base configuration. A system group name can represent multiple systems. This allows SMS to support more than eight systems per SMS complex while retaining the existing configuration format of the configuration data set.

A system group name matches a Parallel Sysplex name and refers to all systems defined as part of the Parallel Sysplex that are:

- Running the same SMS configuration
- Defined in the configuration using the name of the Parallel Sysplex to which they belong (that is, system group)
- NOT defined in the configuration by their system names.

For a discussion on running SMS in a Parallel Sysplex and special considerations to observe, see “Running SMS in a Parallel Sysplex Environment” on page 5.

On the ISMF panel, the System Name and Sys Group fields display, in alphabetic order, the systems or system groups that you have previously defined. You must add at least one system name or system group name to a base configuration before you can save it in the SCDS, and you can define up to 32 system names, system group names, or both to the SMS complex. Note that when you add, delete, or rename a system name in a base configuration, you must activate this configuration pointing to an empty COMMDS.

The system name must match the value of the SYSNAME parameter in the IEASYS*yy* member of the SYS1.PARMLIB used on that system. The system group name must match the value of the SYSPLEX parameter in the COUPLE*xx* (XCF) member of the SYS1.PARMLIB used on that system.

- To add a system name or a system group to the SCDS base configuration:

  Choose option 1, Add, and enter the system name in the System Name field or system group name in the Sys Group Name field. Press ENTER to add the system.
- To delete a system or a system group from the base configuration:

  Choose option 2, Delete, and enter the system name in the System Name field or enter the system group name in the Sys Group Name field. Press ENTER to delete the system or system group name from the SCDS. You are asked to confirm that you want the system or system group deleted before it is deleted.
- To rename a system or a system group in a base configuration:

  Choose option 3, Rename, enter the current name of the system in the System Name or Sys Group Name field, and enter the new name of the system in the New System/Sys Group Name field. Press ENTER to rename the system or system group.

  Note: The Rename option cannot be used to change the type of name. To change a system to a system group, or a system group to a system, you must delete the name, then add it as the desired type.

See “Recovering from a Systems Failure in the SMS Complex” on page 211 for information on recovering from a systems failure in the SMS complex.

Defining the Base Configuration for VSAM Record-Level Sharing

In order for DFSMSdfp to use the coupling facility (CF) for VSAM RLS, you must add CF cache structures to the SMS base configuration and define the cache set(s) with which they are associated.

Using Cache Structures

CF cache structures are defined to MVS using MVS coupling facility resource management (CFRM) policies, which determine how and where the structures are allocated. You associate these cache structures with a cache set name in the base configuration. The cache set name is also specified in a storage class definition. When a storage class associated with a data set contains a cache set name, the data set becomes eligible for VSAM record-level sharing and can be placed in a CF cache structure associated with the cache set. The system selects the best cache structure within the cache set defined for the storage class.

CF cache structures must have the same system connectivity as any storage groups that might be assigned to those cache structures. For example, if Storage Class 1 maps to cache set CS1, then the CF structures in CS1 must have the same system connectivity as the storage groups with which Storage Class 1 is associated. Connectivity of CF cache structures to all systems in the Parallel Sysplex simplifies managing and changing the configuration.

Note: In a JES3 environment, be careful to define cache set names only in those SMS storage classes that are used by data sets opened for VSAM RLS processing. When you define a cache set name in a storage class, any job accessing a data set associated with that storage class is scheduled on a VSAM RLS-capable system (one where the SMSVSAM address space has been successfully initialized). If all storage classes have cache set names defined for them, then all jobs accessing SMS-managed data sets are scheduled to VSAM-RLS-capable systems. This could cause a workload imbalance between those systems and down-level systems. The scheduling system must be at least a DFSMS/MVS 1.3 system.

See “Defining CF Cache Structures” on page 235 for more information on defining CF cache structures. See “Defining the Base Configuration” on page 27 for more information on defining the base configuration.

Defining Cache Sets

To define a cache set and specify the cache structures associated with it:

1. Select option 8 from the ISMF Primary Option Menu for Storage Administrators. ISMF displays the Control Data Set (CDS) Application Selection panel, shown in Figure 10 on page 34.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                          CDS APPLICATION SELECTION
Command ===>

To Perform Control Data Set Operations, Specify:
  CDS NAME . . 'USER6.TESTCDS'
                            (1 to 44 Character Data Set Name or 'Active')

Select one of the following Options:

  7  1. Display       - Display the Base Configuration
     2. Define        - Define the Base Configuration
     3. Alter         - Alter the Base Configuration
     4. Validate      - Validate the SCDS
     5. Activate      - Activate the CDS
     6. Cache Display - Display CF Cache Structure Names for all CF Cache Sets
     7. Cache Update  - Define/Alter/Delete CF Cache Sets

If CACHE Display is chosen, Enter CF Cache Set Name . . *
                             (1 to 8 character CF cache set name or * for all)

Use ENTER to Perform Selection;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 10. CDS Application Selection for VSAM RLS

2. Specify the name of the SCDS that is to contain the base configuration for VSAM RLS in the CDS NAME field. On this panel, the CDS NAME field is primed with the last used SCDS.
3. Select option 7, Cache Update, and press ENTER.

   ISMF displays the CF Cache Set Update panel shown in Figure 11.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
DGTDBSA3                     CF CACHE SET UPDATE                 Page 1 of 1
Command ===>

SCDS NAME  : USER6.TESTCDS
Define/Alter/Delete CF Cache Sets:       ( 000  Cache Sets Currently Defined )

 Cache Set                     CF Cache Structure Names
 ________   ________________ ________________ ________________ ________________
            ________________ ________________ ________________ ________________
 ________   ________________ ________________ ________________ ________________
            ________________ ________________ ________________ ________________
 ________   ________________ ________________ ________________ ________________
            ________________ ________________ ________________ ________________
 ________   ________________ ________________ ________________ ________________
            ________________ ________________ ________________ ________________
 ________   ________________ ________________ ________________ ________________
            ________________ ________________ ________________ ________________
 ________   ________________ ________________ ________________ ________________
            ________________ ________________ ________________ ________________
More CF Cache Sets to Add?  . . N  (Y/N)
Use ENTER to Perform Validation; Use UP/DOWN Command to View other Pages;
Use HELP Command for Help; Use END Command to Save and Exit.
```

Figure 11. Cache Set Update Panel for VSAM RLS

4. Specify the name of a cache set. You can define up to 256 cache set names.
5. Specify the names of all CF cache structures associated with the cache set. You can specify up to eight CF cache structures for each cache set.

Cache structures must be previously defined to MVS in a CFRM policy.

Completing the Base Configuration

After defining base configuration attributes, you can verify their completeness and correctness by pressing ENTER. If your SCDS base configuration contains any errors, the cursor moves to the field in the error and an error message appears in the short message area. Correct any errors and press ENTER to verify the new contents of the SCDS base configuration.

After correcting all errors, use the END command to save the SCDS base configuration.

If at any time you want to leave either the SCDS Base Define or Alter panel without saving the changed base configuration information, issue the CANCEL command.

Chapter 4. Defining Storage Groups

When managing non-system-managed DASD volumes, you view and maintain them as individual devices. If you have too many critical data sets on a volume, you have to spread the data sets across other volumes to remove I/O bottlenecks. If you are allocating data sets manually, you might not be using the volume as efficiently as you could. Some volumes containing critical data sets might be under-used. Other volumes with lower activity data sets might be overpopulated and require extra storage administrator attention to ensure that space is preserved for new data set allocations and extensions. In these and other instances, you have to work on individual volumes to solve your problems.

SMS simplifies the management of DASD, mountable optical volumes, and tape volumes by pooling them together in storage groups. This chapter describes storage groups and shows you how to define them using the ISMF Storage Group Application.

Understanding Storage Groups

Storage groups represent the physical storage managed by SMS. This storage can be collections of DASD volumes, volumes in tape libraries, optical devices, or virtual input/output (VIO) storage. A storage group, used with storage classes, separates the logical requirements of accessing data from the physical requirements to store the data. You can use storage group attributes to specify how the system should manage the storage group. You use the storage group ACS routine to assign a new data set or object to a storage group. You can assign multiple candidate storage groups (except for objects), in which case the system chooses a specific storage group from your list. Storage group definitions are not apparent to users. Only you, as storage administrator, can define, alter, or display storage group definitions.

A storage group can be VIO, dummy, pool, object, object backup, or tape. VIO storage groups are not associated with volumes. Dummy storage groups are associated with non-existent volumes. When defining pool storage groups, you must ensure that the actual, physical paths connecting systems to DASD volumes match the desired logical paths specified in the storage group definitions. Merely establishing a physical connection from a system to a DASD volume does not provide access. Within the storage group definition, you must specify which systems have access to which storage groups, and which storage groups have access to which DASD volumes. Likewise, merely defining a system to have access to a DASD volume does not establish a physical connection. You need to ensure that the physical connection exists.

For tape storage groups, one or more tape libraries (and not volumes) are associated with them. Connectivity is defined at both the library level and the storage group level. If a storage group is connected to certain systems, then any libraries associated with that storage group must be connected to the same systems. Scratch volumes are added to storage groups when they are used. Private volumes can be added when they are entered in a library. Private volumes are removed from a storage group and returned to the common scratch pool when they are returned to scratch status.

For a more specific discussion of how to use storage groups with objects, see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support*.

DASD Storage Groups

When you define a pool storage group, you specify the volume serial numbers of the DASD volumes you are including in the storage group rather than their physical addresses. Each DASD volume in a storage group must contain a VSAM volume data set (VVDS) and an indexed volume table of contents (VTOC). The VVDS is automatically created after allocation of the first system-managed data set on a volume, if the VVDS is not already defined.

Two storage groups cannot share a DASD volume. You must define an entire volume to a single pool storage group. Also, a data set can only reside in one pool storage group. A data set can span volumes within a single pool storage group, but it cannot span volumes belonging to several pool storage groups. For VSAM data sets, the entire sphere (base cluster and all alternate indexes) must be in the same storage group.

Although not required, we recommend that you define pool storage groups so that they only contain devices of the same geometry. The device *geometry* is the track size and number of tracks per cylinder for the device.

By defining pool storage groups so that the device geometry is the same for all volumes in the storage group, you can ensure that volumes of the same geometry are available when multivolume data sets need to extend to new volumes.

For example, if you have 3380 and 3390 devices, you should define at least two storage groups: one containing 3380 devices, and another containing 3390 devices.

Because 3390 devices in 3380 track compatibility mode are geometrically the same as 3380 devices, you can combine these devices in a single storage group. Because the 3390 devices are in 3380 track compatibility mode, the access methods see them as 3380 devices.

Although you should separate devices according to geometry, you do not need to separate them according to capacity. For example, you can combine all models of the 3390 into a single storage group. The only effect the different capacities has is on volume thresholds. See *MVS/ESA SML: Managing Storage Groups* for information on selecting appropriate threshold levels.

Devices of the same geometry can have different performance characteristics. These devices coexist in the same storage group, and enhanced volume selection for SMS manages data set placement accordingly. With enhanced volume selection, even devices with vastly different performance characteristics (like the IBM 3390 Model 3 and the IBM 3995 Model 153) can reside in the same storage group. Note that the 3995-153 does have a non-zero initial access response time.

Tracking DASD Volume Status for Data Sets

To prepare new DASD volumes for SMS storage groups, use the Device Support Facilities (ICKDSF) INIT command to assign new DASD volume serial numbers and allocate an indexed VTOC. You can use the STORAGEGROUP (STGR) keyword of the INIT command to make a DASD volume available for allocation of new system-managed data sets.

You can use INIT from the ISMF Volume Application and the Storage Group (through LISTVOL) Application. If you go to the Volume Application directly from the

Primary Option Menu, you initialize the DASD volume as a non-system-managed volume. If, instead, you work from a DASD volume list generated within the Storage Group Application, you initialize the DASD volume as a system-managed volume. When new volumes are added to a storage group in the active configuration, we recommend explicitly allocating small temporary data sets to each new volume to update the SMS volume space statistics.

As you bring DASD volumes under the control of SMS, you need to keep track of those that are ready to be system-managed. The physical DASD volume status can be one of the following: converted, initial, non-SMS, or unknown. To display this information, on the Storage Group Application panel, enter the LISTVOL command against pool storage groups.

Convert status indicates that the DASD volume is converted and is fully available for SMS control. To be converted, all data sets on the DASD volume must have an associated storage class, and all permanent data sets on the DASD volume must be cataloged in an integrated catalog. The DASD volume must have a VTOC index.

Initial status indicates that an attempt to convert the DASD volume has been made, but the DASD volume contains data sets that fail to satisfy the requirements of a converted volume. In the initial state, no new allocations can be made on the DASD volume. You cannot extend data sets to additional DASD volumes. The initial status allows DFSMSdss to process the DASD volume without having to issue a RESERVE for the entire duration of the conversion process. Also, to be eligible for conversion, data sets and the DASD volume must have an organization that is supported by SMS. For more information, refer to *DFSMS/MVS Implementing System-Managed Storage*.

Object and Object Backup Storage Groups

An object storage group is a storage group that defines the physical storage used for objects. For objects, the storage group allows you to define an object storage hierarchy. An object storage hierarchy consists of an object directory and one or more of the following:

- A DB2 object storage table
- Library-resident optical volumes
- Shelf-resident optical volumes
- Tape volumes

You can define an object storage group that does not include an optical library. If an object storage group does not include an optical library, the objects reside on DASD volumes or tape volumes and do not get moved to optical volumes.

An object can move up and down the hierarchy within a single storage group but cannot move outside of that group. Objects move within the storage hierarchy depending initially on management class and storage class criteria for the object collection, and subsequently on ACS routines.

You can define two types of storage groups for the object access method (OAM): object and object backup. An object storage group defines an object storage hierarchy. The object backup storage group defines the optical and tape devices used for backing up objects. A system within an SMS complex can have multiple object storage groups, but only one object backup storage group. As of DFSMS/MVS 1.5, an object or object backup storage group can be enabled by and connected to more than one system in a Parallel Sysplex. There is no limit on the number of object storage groups defined by each system in the SMS complex. Additionally, you can define the high-level qualifier (HLQ) associated with the object storage groups.

Two timing attributes are associated with object storage groups: Cycle Start Time and Cycle End Time. These attributes define a window during which the OAM Storage Management Component (OSMC) can automatically start its storage management processing for the storage group. The window occurs once each day. An object backup storage group has no timing attributes.

You can also specify an OSMC processing system name, which identifies which OAM system in the sysplex performs OSMC processing for this object storage group. This prevents multiple systems from trying to process the same storage group at the same time, especially when the Cycle Start Time is used for automatically starting storage group processing.

For more information about managing objects, see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support*.

Defining Real and Pseudo Optical Libraries in Object Storage Groups

When defining an object storage group, you can specify any number of real optical libraries *or* any number of pseudo optical libraries. OAM stores an object using any optical drive and optical volume associated with any of the optical libraries listed in the object storage group definition.

The object or object backup storage groups can share optical libraries. A single optical library can contain optical volumes belonging to several different object storage groups. There can be multiple object storage groups per system. Each object storage group on a system must have a unique qualifier that identifies a DB2 database containing the object directory.

Defining Optical Libraries in Object Backup Storage Groups

When defining an object backup storage group, you can specify one to eight real optical libraries, *or* one to eight pseudo optical libraries, *or* you do not need to specify any optical libraries.

You can define multiple libraries within a given SCDS. You can associate multiple libraries with an object backup storage group by specifying multiple library names on the Object Backup Storage Group Define or Alter panels.

OAM stores the backup copy of the object using any optical drive and optical volume associated with any one of the optical libraries listed in the object backup storage group definition.

Writing Objects

Objects can be written using any drive in any library associated with the object storage group, using any volume assigned to that storage group, as long as there is enough space to satisfy the request, and the volume is write-compatible with the drive. If no group volumes are available, the request is satisfied using a scratch volume residing in a library associated with the object storage group.

The storage class for an object determines where an object is written (DASD, optical, or tape). Objects can be written to tape using the media type and tape unit name defined in the SETOAM statements in the CBROAMxx member of PARMLIB.

Both object and object backup storage groups have three attributes concerned with writing objects:

- The drive startup threshold
- The volume full threshold
- The mark volume full on first write failure option

The *drive startup threshold* is the maximum number of write requests that can be waiting for each drive that is currently processing write requests for the storage group. If the threshold is specified and then exceeded, an attempt is made to start another drive to process write requests to the storage group.

When the number of free KB on an optical volume falls below the *volume full threshold*, the volume is marked full. If you specified *mark volume full on first write failure*, an optical volume is marked as full the first time an attempt to write an object on the optical volume fails because not enough space remains on the optical volume.

A mountable optical disk has a volume on each side. Each volume is referred to by a separate volume serial number. Each volume contains a volume label, an optical volume table of contents (OVTOC), and a user object data area. OAM dynamically allocates a new volume whenever the current volume is full and there is another request to write to the same storage group. OAM only uses dynamic allocation for tape volumes.

For more information about writing objects, see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support*.

OAM Collection Names

OAM collection names provide another level of grouping called a *collection*. A collection is a group of objects that share a common storage group and have the same default initial storage class and management class attributes. All objects within a collection must have unique names. However, you can have two objects with identical names if the objects belong to separate collections.

OAM collections are cataloged in integrated catalog facility (ICF) catalogs using OAM collection names. An individual object does not have an entry in the catalog, but is cataloged using the name of the collection which it is a member of.

Each object is a member of a collection and each collection is part of one storage group. A collection does *not* span storage groups. OAM identifies an object by its collection name and its object name. An object is described by an entry in a DB2 object directory. A collection is described by an MVS collection name catalog entry and a corresponding OAM collection identifier table entry.

OAM provides two primary interfaces for processing objects: the application program interface (OSREQ) and the OAM Storage Management Component (OSMC). Both address objects using the object's collection name and object name including those cases where the object must be identified to the ACS routines for the STORE (OSREQ), CHANGE (OSREQ), and CTRANS (OSMC class transition) environments. Refer to *DFSMS/MVS OAM Planning, Installation, and Storage*

Administration Guide for Object Support and *DFSMS/MVS OAM Application Programmer's Reference* for more information.

Tape Storage Groups

The Storage Group Application supports a tape storage group type for tape libraries. The LISTVOL line operator on the Storage Group List supports the mountable tape volume list.

The storage administrator can specify tape as the storage group type and can then define and alter tape storage groups and their relationships to the systems in the SMS complex. These relationships are NOTCON, ENABLE, DISALL, DISNEW, QUIALL, or QUINEW. See "Defining System Access to Pool and VIO Storage Groups" on page 47 for an explanation of these relationships.

A tape storage group is a collection of tape cartridges and is associated with one to eight tape libraries. A scratch volume is added to a storage group on use. A private volume can be added to a storage group when it is entered in a library. You can direct allocations to a local or remote library or to a specific library by assigning the appropriate storage group in the storage group ACS routine.

Library Definition

Tape libraries are supported only when the system is IPLed using MVS/Hardware Configuration Dialog (MVS/HCD) generated input/output definition file (IODF). If MVS/HCD is not currently being used, the installation must convert from MVS configuration program (MVSCP) to MVS/HCD and re-IPL in order to use a tape library and exploit tape library support.

SMS

For SMS support of the tape library, an SCDS of the proper level must be activated. This SCDS must contain appropriate library definitions and tape storage group definitions. Information that the storage administrator must provide specifically for an Automated Tape Library Data Server (ATLDS) is listed below:

- Data compaction, media type and recording technology information can be included in data classes that are used in allocation of devices in a tape library.
- New storage groups of type TAPE must be defined. The storage groups are involved in the allocation of new data sets to volumes within a tape library.
- A tape library must be defined in the SCDS.
- Default data class defined to the library.
- ACS routines must specify a storage class and the storage group selected must be associated with a tape library.

Planning Storage Groups for Data Sets

Ideally, you would have only one storage group containing all of your data sets, and the system would manage everything. Realistically, you have to account for a variety of data sets, such as databases, large data sets, temporary data sets, and tape data sets.

Before you actually define your storage groups, you should gather the following information:

- Existing I/O hardware configuration

- Projected hardware requirements
- Estimated general requirements of user groups
- Anticipated security requirements
- Backup and recovery requirements
- Data set sizes
- Required system access, based on both shared data requirements and the needs of user groups
- The number of systems to which the storage group is connected.

To begin your planning, define one primary storage group and identify all the data sets that do not fit into this general storage group category. Keep in mind your storage class and management class requirements.

For data sets that require continuous access (continuous for the storage class availability attribute) the storage group must contain sufficient volumes that do not interrupt data availability for a single device failure (for example, dual copy or array DASD).

The highest performance is generally obtainable through use of cache at the subsystem level, the device level, or both. Sufficient volumes with cache access should be available in a storage group for data sets that require high performance. Data sets that have primarily write access (i.e., write BIAS in the storage class) benefit most from a DASD fast write capability.

| For data sets requiring concurrent copy, ensure that sufficient volumes in the
| storage group are attached through 3990 Storage Controls with Extended Platform,
| or through an IBM RAMAC Virtual Array.

For striped data sets, ensure that there are a sufficient number of separate paths to DASD volumes in the storage group to allow each stripe to be accessible through a different path. The maximum number of stripes for one data set is 16. (A stripe is the portion of a striped data set that resides on one volume.)

Figure 12 shows the formula that is used to derive the optimum number of stripes that are allocated to the data set:

```
Sustained data rate / Device transfer rate = Optimum number of stripes
```

Figure 12. Formula for Deriving Optimum Number of Stripes

where:

- Sustained data rate is the target throughput rate, specified in the storage class definition
- Device transfer rate is based on the published characteristics of the device

Ideally, the number of stripes should be on different paths, however, if this is not possible, SMS allocates to a smaller number of stripes.

If a data set has a management class that specifies automatic backup or migration, you must direct the data set to a storage group that is eligible to be processed for automatic backup or migration. You need to specify through ISMF the system or system group that is to perform the backup or migration only if that particular system or system group should do the processing. Otherwise, do not specify a particular system or system group to do the processing. It is done automatically. Management class attributes do not apply to tape data sets, so only assign data sets to tape which do not require management class services.

See *MVS/ESA SML: Managing Storage Groups* for more information about planning and implementing storage groups.

Planning Storage Groups for OAM Object Collections

For each major application that processes objects, one or more object collections must be defined to provide for the storing, cataloging, and retrieval of objects used by that application. Each object collection must be assigned to a storage group by the storage group ACS routine. Multiple collections can be assigned to the same storage group.

A storage group for OAM object collections provides a storage hierarchy containing a DB2 table for the object directory for objects in the storage group, DB2 tables for storage of objects on DASD, and optical volumes for the storage of objects on optical media.

| If an Initial Access Response Seconds (IARS) attribute that is not equal to 0 and
| less than 3 is specified in the storage class, and an optical volume is available,
| OAM satisfies the object store request.

Before you actually define your storage groups, you should gather the following information:

- Existing I/O hardware configuration
- Projected hardware requirements
- Estimated general requirements of user groups
- Anticipated security requirements
- Backup and recovery requirements
- Object sizes
- Required system access, based on both shared data requirements and the needs of user groups
- The number of systems to which the storage group is connected.

For more information on planning for objects, see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support*.

Defining Storage Group Attributes

You can use ISMF to define your storage groups by selecting option 6, Storage Group, from the ISMF Primary Option Menu for storage administrators. Figure 13 on page 45 shows the Storage Group Application Selection panel.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                     STORAGE GROUP APPLICATION SELECTION
Command ===>

To perform Storage Group Operations, Specify:
  CDS Name  . . . . . . . 'SMS.SCDS1.SCDS'
                               (1 to 44 character data set name or 'Active' )
  Storage Group Name  . . VI01              (For Storage Group List, fully or
                                             partially specified or * for all)
  Storage Group Type  . . VIO               (VIO, POOL, DUMMY, OBJECT, OBJECT
                                             BACKUP, or TAPE)

Select one of the following options  :
  2  1. List    - Generate a list of Storage Groups
     2. Define  - Define a Storage Group
     3. Alter   - Alter a Storage Group
     4. Volume  - Display, Define, Alter or Delete Volume Information

If List Option is chosen,
   Enter "/" to select option      Respecify View Criteria
                                   Respecify Sort Criteria
Use ENTER to Perform Selection;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 13. Defining a Storage Group

To define a storage group, you must specify a control data set name, storage group name, and storage group type on the panel and select option 2, Define. The control data set name must be the name of an SCDS. ISMF primes the field with the name last used within ISMF. (The default is 'active', which represents the currently active configuration, but you cannot define or alter the storage groups for the 'active' configuration.)

In the Storage Group Name field, you must specify the name of the storage group that you are defining. ISMF primes the field with the name last used within ISMF. The default is an asterisk, *, which represents all storage groups in the specified control data set name.

In the Storage Group Type field, you must specify one of the following types:

VIO
Virtual I/O (VIO) storage groups are used to allocate data sets to VIO, which simulates the activity of a DASD volume. VIO storage groups do not contain any actual DASD volumes. You can put temporary data sets in VIO storage groups.

Pool
Pool storage groups contain the volume serial numbers of system-managed DASD volumes. Pool storage groups can be used for both temporary and permanent data sets.

Dummy
Dummy storage groups contain the volume serial numbers of DASD volumes that no longer reside on the system but that you want to treat as SMS DASD volumes. Using dummy storage groups allows existing JCL that explicitly references the DASD volumes in VOL=SER statements to work. If end users specify a VOL=SER in their JCL, and that volume serial number is in a dummy storage group list, then SMS issues a catalog request to find the desired data set rather than using the volume serial number.

Volumes in dummy storage groups cannot be used when performing volume allocations. For example, the following DD statement, where DUMMY1 is a volume in a dummy storage group, does not work:

```
//DD1  DD VOL=SER=DUMMY1,UNIT=SYSDA,DISP=SHR
```

A dummy storage group should not contain the volume serial number of a DASD volume that exists in the system. If the DASD volume exists in the system and the data set is system-managed, no JCL errors occur but the job fails during allocation. If the DASD volume exists in the system and the data set is not system-managed, then the resulting errors depend on the type of data set. For uncataloged data sets, either the data set cannot be found or the wrong data set with the same name is found. For cataloged data sets, the job fails during allocation.

Object
Object storage groups identify an object storage hierarchy.

Object Backup
Object backup storage groups define the group that is to be used to contain backup data and can specify one or more optical libraries that can contain backup copies of objects. There can be only one object backup storage group per system.

The object backup storage group can be defined with no optical libraries and have the backup copies of objects written to tape by specifying SETOAM statements in the CBROAMxx member of PARMLIB.

Tape Tape storage groups identify storage groups to maintain system-managed tape volumes.

After specifying a storage group type, select option 2, Define, and press ENTER to reach the appropriate Storage Group Define panel.

The remainder of this chapter provides the panel sequences in the Storage Group Application and explains the attribute values for each type of storage group. You can leave any of the Storage Group Define panels at any time without saving the storage group by issuing the CANCEL command.

Defining a VIO Storage Group

Figure 14 on page 47 shows the VIO Storage Group Define panel.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                         VIO STORAGE GROUP DEFINE            Page 1 of 2
Command ===>

SCDS Name  . . . . : SMS.SCDS1.SCDS
Storage Group Name : VIO1

To DEFINE Storage Group, Specify:
 Description ==>
             ==>
 VIO Maxsize . .                   (8 to 2000000)
 VIO Unit  . . .                   (DEVICE TYPE)

 System/Sys                    System/Sys
 Group Name      Status        Group Name      Status
 ----------      ------        ----------      ------
 SYS1       ===> ENABLE                   ===>          * SYS GROUP = sysplex
            ===>                          ===>            minus systems in the
            ===>                          ===>            sysplex explicitly
            ===>                          ===>            defined in the SCDS
            ===>                          ===>
Use ENTER to Perform Verification; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 14. Defining VIO Storage Group Attributes

The SCDS Name and Storage Group Name fields are output fields that contain the SCDS and storage group names you specified in the Storage Group Application Selection panel. The Description field is an optional field of 120 characters in which you can describe the VIO storage group.

To define a VIO storage group, you must specify a data set maximum size for VIO and a device type. You must also assign a storage group status for each system and system group.

After specifying the attribute values and system status relationships, issue the END command. END saves the newly defined VIO storage group and returns you to the Storage Group Application Selection panel.

VIO Maxsize

VIO Maxsize is a required value that represents the maximum size, in KB, of data sets to be allocated to VIO. If a data set exceeds the maximum size, then the allocation fails. However, if the storage group ACS routine assigns both a VIO storage group and a pool storage group to the list of candidate storage groups for the data set, and the data set exceeds the maximum size, then the data set is allocated to the pool storage group.

VIO Unit

VIO Unit is a required field that represents the generic DASD device type (for example, 3380 or 3390) that this storage group simulates.

Defining System Access to Pool and VIO Storage Groups

If you want a system to access a pool storage group, you must ensure that devices on which the groups reside are physically connected to the system.

No devices are associated with a VIO storage group. However, the SMS status associated with the VIO storage group determines if this storage group can be used by each system in the complex. The SMS statuses, described below, show the relationships between each system in the SMS complex and a given VIO or pool storage group.

DISALL
: Disable All prevents the system from allocating or accessing data sets in the storage group.

DISNEW
: Disable New prevents the system from allocating new data sets (and DISP=MOD data sets that do not currently exist) in the storage group. DISNEW prevents data sets from extending to a volume whose volume or storage group status is DISNEW.

ENABLE
: The system can allocate and access data sets in the storage group. ENABLE is the default relationship between a system and a storage group.

NOTCON
: Not Connected indicates that the storage group is defined but not accessible to the system. It resembles DISALL, except you cannot dynamically change the NOTCON system status. If you want a storage group to have access to NOTCON volumes, physically connect them to the system, define the connection by changing the system status in the ISMF Storage Group Application, and activate the configuration that defines the connection.

QUIALL
: Quiesce All prevents the system from scheduling jobs that allocate or access data sets in the storage group. This state only affects JES3 systems, that support job scheduling.

When the job is executing, QUIALL has the same effect in a JES2 or JES3 environment. In an JES2 environment, the volumes become a secondary volume selection candidate through QUIALL. *Secondary volume selection candidacy* means that these volumes are still available for allocation, but less preferred than other volumes.

For more information about primary and secondary volume selection candidacy, see "Conventional Volume Selection" on page 105.

QUINEW
: Quiesce New prevents the system from scheduling jobs that allocate new data sets (and DISP=MOD data sets that do not currently exist) in the storage group.

If the system uses JES3 to schedule jobs, you can use QUINEW only as long as other volumes are available. Before scheduling a job, JES3 verifies that all resources are available, unlike JES2 which schedules a job even if all the needed resources are not available. This can lead to a contention for resources, since a job running under JES2 can hold some resources while it waits for others to become available. Under JES3, if any of the candidate volumes are available, JES3 schedules the job and SMS selects the volume. Under both JES2 and JES3, SMS selects QUINEW volumes only as a last resort.

For more information about primary and secondary volume selection candidacy, see "Conventional Volume Selection" on page 105.

Defining a Dummy Storage Group

Figure 15 shows the Dummy Storage Group Define panel.
SCDS Name and Storage Group Name are output fields containing the SCDS and

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                         DUMMY STORAGE GROUP DEFINE
Command ===>

SCDS Name . . . . . : SMS.SCDS1.SCDS
Storage Group Name  : DUMMY1

To DEFINE Storage Group, Specify:

 Description ==>
             ==>












Use ENTER to Perform Verification;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 15. Defining Dummy Storage Group Attributes

storage group names that you specified in the Storage Group Application Selection panel.

Description is an optional field of 120 characters where you can describe the dummy storage group. Issue the END command to save the newly defined dummy storage group and return to the Storage Group Application Selection panel.

Initially, the storage group contains no volume serial numbers. To add volume serial numbers to the storage group, select option 4, Volume, from the Storage Group Application Selection panel shown in Figure 13 on page 45. ISMF primes the other fields on the panel with the most recently specified values. (You do not need to specify a Storage Group Type.) Then, press ENTER to define volume serial numbers to the storage group in the Storage Group Volume Selection panel shown in Figure 16 on page 50.

On this panel, you specify the volume serial numbers that you want to belong to the dummy storage group.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                        STORAGE GROUP VOLUME SELECTION
Command ===>

CDS Name  . . . . . : SMS.SCDS1.SCDS
Storage Group Name  : DUMMY1
Storage Group Type  : DUMMY

Select One of the following Options:
  2  1. Display      - Display SMS Volume Statuses (Pool only)
     2. Define       - Add Volumes to Volume Serial Number List
     3. Alter        - Alter Volume Statuses   (Pool only)
     4. Delete       - Delete Volumes from Volume Serial Number List

Specify a Single Volume (in Prefix), or Range of Volumes:
      Prefix   From     To    Suffix  Hex
      ______  ______  ______  _____    _
 ===> SYS     001     077                    ('X' in HEX field allows
 ===> DFPIP1                                  FROM - TO range to include
 ===> SYS     25      30      A        X      hex values A through F.)
 ===>         001     010     VOL
Use ENTER to Perform Selection;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 16. Defining Dummy Storage Group Volume Serial Numbers

To add DASD volumes to the dummy storage group, select option 2, Define, and specify the volume serial numbers in the Specify a Single Volume (in Prefix), or Range of Volumes field.

You can specify a single volume serial number by typing the number under Prefix. The value can be from one to six characters, but it must be a fully specified volume serial number. You can specify a range of volume serial numbers by typing the prefix or suffix that is common to a set of volumes under Prefix or Suffix, the low individual volume number under From, and the high individual volume number under To.

You can specify an X under Hex to include hexadecimal values in your range. For example, in Figure 16 volume numbers SYS001 through SYS077 are specified on the first volume specification line. Volume number DFPIP1 is specified on the second line. Volume numbers SYS25A through SYS30A, including SYS2AA through SYS2FA, are specified on the third line. Volume numbers 001VOL through 010VOL are specified on the fourth line. You can add as many as a hundred volume serial numbers at a time to a pool storage group this way.

After you specify volume serial numbers, press ENTER to add the DASD volumes to the dummy storage group. Define any additional volume serial numbers, then use END to return to the Storage Group Application Selection panel. If any volumes cannot be added, a list of the volumes that could not be added is displayed.

When adding a new volume serial number (volser), you must make sure that all volsers defined in the volume list are unique. ISMF verification only verifies that all volsers are unique within the same dummy or pool storage group. No attempts are made to verify the device types of each volume. Therefore, if a library volser is duplicated in a dummy or pool storage group, this error is not detected until a request is issued to mount this volser.

Defining a Pool Storage Group

Figure 17 shows the Pool Storage Group Define panel.
SCDS Name and Storage Group Name are output fields containing the SCDS and

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                         POOL STORAGE GROUP DEFINE
Command ===>

SCDS Name . . . . . : SMS.SCDS1.SCDS
Storage Group Name  : POOL1
To DEFINE Storage Group, Specify:
 Description ==>
             ==>
 Auto Migrate . . Y  (Y, N, I or P)   Migrate Sys/Sys Group Name . .
 Auto Backup  . . Y  (Y or N)         Backup Sys/Sys Group Name  . .
 Auto Dump  . . . N  (Y or N)         Dump Sys/Sys Group Name  . . .

 Dump Class . . .                     (1 to 8 characters)
 Dump Class . . .                     Dump Class . . .
 Dump Class . . .                     Dump Class . . .

 Allocation/migration Threshold: High . . 85  (1-99)      Low  . .     (0-99)
 Guaranteed Backup Frequency  . . . . . .           (1 to 9999 or NOLIMIT)

 DEFINE SMS Storage Group Status . . .... N   (Y or N)
Use ENTER to Perform Verification and Selection;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 17. Defining Pool Storage Group Attributes

storage group names that you specified in the Storage Group Application Selection panel.

Description is an optional field of 120 characters where you can describe the pool storage group.

The storage group and management class are interrelated. The storage group Auto Migrate and Auto Backup parameters specify whether the volumes in this storage group are eligible to be processed automatically. The management class, assigned to the data sets residing on the volumes, determines whether and how to process the data sets on the volume. In contrast, if you set Auto Migrate or Auto Backup to NO in the storage group, the volumes in the storage group are not processed and data sets residing in the storage group are not migrated or backed up.

Auto Migrate

In Auto Migrate, you specify whether you want the DASD volumes in this storage group to be eligible for automatic space management processing. It is a required field, which ISMF primes with the value **Y**, yes. **Y** specifies that DFSMShsm is to perform automatic space management processing if processing windows are defined to DFSMShsm. Data sets are eligible for primary space management. If the Setsys Intervalmigration attribute has been specified in DFSMShsm, the data sets are also eligible for interval migration. The value **N**, no, specifies that data sets are not eligible for automatic migration.

I specifies that data sets are eligible for primary space management and specifies that DFSMShsm is to perform automatic interval migration independent of the DFSMShsm Setsys Intervalmigration option. **I** also causes DFSMShsm to perform the same functions as if **Y** had been specified. Interval migration is performed on only those volumes whose occupancy is at or above their high threshold. The most frequent DFSMShsm can do interval migration is hourly. If you select **I** with a low threshold value of 0, DFSMShsm migrates all the eligible data sets in the selected storage group. **I** is most useful for storage groups used with tape mount management.

The value **P** specifies that data sets are eligible for primary space management but interval migration is not performed even if Setsys Intervalmigration is specified.

If for some reason (such as a system outage) system-managed temporary data sets are not deleted at the end of a job, DFSMShsm attempts to delete them during primary space management. In order for DFSMShsm to delete these data sets, be sure to specify Auto Migrate as **Y** or **I**, or Auto Backup as **Y**.

Auto Backup

In Auto Backup, you specify whether you want the DASD volumes in this storage group to be eligible for automatic backup processing. It is a required field, which ISMF primes with the value Y, yes. Y specifies that DFSMShsm is to back up the data sets on the volume according to management class requirements.

Auto Dump

In Auto Dump, you specify whether you want to automatically dump all the DASD volumes in this storage group. It is an optional field, which ISMF primes with the value N, no. If you specify a value of Y, yes, DFSMShsm automatically dumps the volumes.

Migrate, Backup, and Dump Sys/Sys Group Name

ISMF no longer verifies that specified system or system group names are defined to the base configuration. You can specify either a system or a system group name in these fields but a specific system specified might not be defined in the configuration since it might be defined as part of a system group. Therefore, if these fields are non-blank, you must take special care to ensure that the values are correct. If the values are incorrect, it could result in the expected DFSMShsm operation not occurring.

DFSMShsm has the capability of processing each storage group for automatic space management, data availability management, and automatic dump processing. A system is eligible to perform the processing when any of the following are conditions are met:

- The name is blank, meaning any system can perform it
- The name specified is the name of the specific host system
- The name specified is the name of the system group to which the system belongs and the system is not individually defined by its system name in the configuration.

DFSMShsm ignores storage groups for which you specify a different system name, and does not process DASD volumes that have already been processed. Do not specify a Sys/Sys Group Name unless processing of the storage group for the function must be performed only on that one host because that limits the capabilities of DFSMShsm to perform the request.

The same rules apply to Backup Sys/Sys Group Name for data availability management processing, and to Dump Sys/Sys Group Name for automatic dump processing. All three Sys/Sys Group Name fields are optional and primed with blanks.

Dump class

In the Dump Class fields, you can specify up to five unique dump class names. When DFSMShsm dumps DASD volumes that belong to the storage group, it directs their contents to the dump classes. To use dump classes, you must define their names and parameters using DFSMShsm. Then on this panel, you can identify the dump class names to SMS. These are optional fields that are primed with blanks.

Migration and Allocation Thresholds

In Allocation/Migration Threshold, you specify an upper and lower space limit for the DASD volumes in a pool storage group. SMS tries to stay within these values by looking at a data set’s primary space allocation before assigning it to a given DASD volume. For example, the SMS volume selection function attempts to prevent allocation of a data set to a given DASD volume if that allocation causes the volume’s high threshold to be exceeded. In addition, this high threshold value is used by DFSMShsm to determine whether data sets should be migrated off a DASD volume in the storage group. The low threshold value is used as the threshold goal in reducing the amount of space occupied on a DASD volume in the storage group during interval migration or daily space management. The low threshold value must be less than or equal to the high threshold value.

Both numbers are percentages of the total space on the DASD volume. If you specify **Y** for Auto Migrate, then you must specify both a high and low threshold. If you specify Y, the low threshold limit is 1. If you specify **I** for automatic interval migration, you can specify a low threshold value of 0 to migrate all the data sets in the selected storage group. The hourly migration trigger for storage groups with a value of AM=I is the occupancy at or above the average of high and low thresholds. Since storage groups used for tape mount management tend to fill up several times a day, allowing interval migration for these storage groups allows DFSMShsm to better keep up with the demand.

ISMF requires that you enter a high threshold value when you specify **N** for Auto Migrate, when defining a pool storage group. Since SMS needs the value for allocation purposes, the High Threshold field is a required field with a primed value of 85.

For more information on specifying allocation and migration thresholds, see *DFSMS/MVS DFSMShsm Storage Administration Reference*.

Guaranteed Backup Frequency

In the Guaranteed Backup Frequency field, you can specify the number of days within the last backup period in which the backup process should have a copy of each of the data sets within the applicable storage group. If Auto Backup is specified as Yes, this attribute is required, otherwise it is optional.

You specify the maximum number of days that can elapse between backups. You can specify from 1 to 9999 days or you can specify NOLIMIT. If you specify NOLIMIT, then data sets in the storage group are backed up according to management class specifications. There is no default.

Defining Pool Storage Group Status

In the SMS Storage Group Status Define panel, indicate whether you want to change the status of the pool storage group with respect to a given system in the SMS complex. Initially, all of the status fields are ENABLE. Specify Y, yes, to change a status. Accept the default N, no, to leave all the status fields as ENABLE.

When you have completed all your entries in the Pool Storage Group Define panel, press ENTER. If you specified Y for the Define SMS Storage Group Status attribute, you get the SMS Storage Group Status Define panel shown in Figure 18, where you can define the status of the storage group to each system in the SMS complex.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                      SMS STORAGE GROUP STATUS DEFINE          Page 1 of 2
Command ===>

SCDS Name  . . . . : SMS.SCDS1.SCDS
Storage Group Name : POOL1
Storage Group Type : POOL
To DEFINE Storage Group System/                           ( Possible SMS SG
Sys Group Status, Specify:                              Status for each:
                                                        - Pool SG Type
 System/Sys      SMS SG    System/Sys      SMS SG         NOTCON, ENABLE
 Group Name      Status    Group Name      Status         DISALL, DISNEW
 ----------      ------    ----------      ------         QUIALL, QUINEW
 SYSTEM1    ===> ENABLE    SYSTEM2    ===> ENABLE       - Tape SG Type
 SYSTEM3    ===> ENABLE    SYSTEM4    ===> ENABLE         NOTCON, ENABLE,
 SYSTEM5    ===> ENABLE    SYSTEM6    ===> NOTCON         DISALL, DISNEW )
 *SYSPLX1   ===> ENABLE               ===>
            ===>                      ===>              * SYS GROUP = sysplex
            ===>                      ===>                minus Systems in the
            ===>                      ===>                Sysplex explicitly
            ===>                      ===>                defined in the SCDS
Use ENTER to Perform Verification; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 18. Defining Pool Storage Group System Status

The system or system group names you defined in the base configuration are listed under System/Sys Group Name. SMS SG Status lists the relationship between the storage group and each system in the SMS complex. You can specify the statuses, which are explained in "Defining System Access to Pool and VIO Storage Groups" on page 47.

After establishing the relationships between your storage group and the systems in the SMS complex, you need to define DASD volumes to the storage group. From the SMS Storage Group Status Define panel, enter the END command to save your values and return to the Pool Storage Group Define panel. On this panel, enter the END command to save the pool storage group. This returns you to the Storage Group Application Selection panel.

Adding Volumes And Defining the SMS Volume Status

On Figure 13 on page 45 select option 4, Volume, and press ENTER to view the Storage Group Volume Selection panel which appears in Figure 16 on page 50.

Select option 2, Define, and specify the volume serial numbers of the DASD volumes that you want to add to the pool storage group. Each time you press enter, you see the SMS Volume Status Define panel shown in Figure 19.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                          SMS VOLUME STATUS DEFINE              Page 1 of 2
Command ===>

SCDS Name . . . . . . : SMS.SCDS1.SCDS
Storage Group Name  . : POOL1
Volume Serial Numbers : 010000 - 010010

To DEFINE SMS Volume Status, Specify:

 System/Sys      SMS Vol     System/Sys      SMS Vol    ( Possible SMS Vol
 Group Name      Status      Group Name      Status       Status for each:
 ----------      -------     ----------      -------      NOTCON, ENABLE,
 SYSTEM1    ===> ENABLE      SYSTEM2    ===> ENABLE       DISALL, DISNEW,
 SYSTEM3    ===> ENABLE      SYSTEM4    ===> ENABLE       QUIALL, QUINEW )
 SYSTEM5    ===> ENABLE      SYSTEM6    ===> ENABLE
 *SYSPLX1   ===> ENABLE                 ===>            * SYS GROUP = sysplex
            ===>                        ===>              minus systems in the
            ===>                        ===>              sysplex explicitly
            ===>                        ===>              defined in the SCDS
            ===>                        ===>
Use ENTER to Perform Verification; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 19. Defining Volume System Status

In the SMS Volume Status Define panel, you define the relationship between the DASD volume in the pool storage group and each system in the SMS complex. The volumes defined to the storage group are listed in Volume Serial Numbers and the systems you defined in the base configuration are listed under Systems.

SMS Volume Status lists the relationship between a DASD volume and each system in the SMS complex. DASD volumes have both a physical (actual) and a logical (system-defined) connection to systems in the SMS complex. You are responsible for maintaining consistency between these two types of connections.

For a system to have access to a data set, you need to define one of the five types of access from the system to a storage group, and you need to define one of the five types of access from the storage group to a DASD volume (ENABLE, DISALL, DISNEW, QUIALL, and QUINEW for both types of access). You must ensure that a physical connection exists from the DASD volume to the system, and the MVS status of the DASD volume must be ONLINE.

See “Defining System Access to Pool and VIO Storage Groups” on page 47 for an explanation of the SMS statuses. The volume statuses show the relationships between each system in the SMS complex and a given DASD volume in the pool storage group.

When a system attempts to allocate a data set, it proceeds in order through the following checks with respect to the current system:

1. Storage group status—ENABLE/DISALL/DISNEW/QUIALL/QUINEW/NOTCON
2. SMS volume status—ENABLE/DISALL/DISNEW/QUIALL/QUINEW/NOTCON
3. MVS volume status—ONLINE/OFFLINE

After defining the status of the DASD volume, issue the END command to return to the Storage Group Application Selection panel. You can define additional DASD volumes to the pool storage group, if you wish.

Assigning DASD Storage Groups to Data Sets

Storage groups can only be assigned through the storage group ACS routine. For a given data set, the storage group ACS routine is executed only if the storage class ACS routine assigns a valid storage class. If the storage class is not valid, allocation fails. If no storage class is assigned, then the data set is not system-managed and is allocated according to the rules in a non-system-managed environment.

You can assign candidate storage groups to a data set allocation from which SMS selects eligible DASD volumes for the data sets. If the storage group ACS routine does not determine a storage group for a data set, allocation fails.

Defining an Object Storage Group

Before you define an object storage group that uses optical volumes, you should define your optical libraries. Please refer to *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support* for more information.

Figure 20 shows the Object Storage Group Define panel.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                        OBJECT STORAGE GROUP DEFINE
Command ===>

SCDS Name . . . . . : USER8.TEST.SCDS
Storage Group Name  : SGO

To DEFINE Storage Group, Specify:
  Description ==>
             ==>
  Qualifier  . . . . .          (1 to 8 character qualifier)
  Cycle Start Time . .       (0-23 or NONE)  End Time . .     (0-23 or blank)
  OSMC Processing System . . . .          (? for list of OSMC System names)

  Library Names   (1 to 8 Characters each):
   ===>           ===>           ===>           ===>
   ===>           ===>           ===>           ===>
  Volume Full Threshold  . . . .      (0-9999)
  Drive Start Threshold  . . . .      (0-9999)
  Volume Full at Write Error . .      (Y or N)
   DEFINE SMS Storage Group Status . .... Y  (DEFINE - Y, ALTER - Y or N)
Use ENTER to Perform Verification and Selection;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 20. Defining Object Storage Group Attributes

The SCDS Name and Storage Group Name fields are output fields containing the SCDS and storage group names that you specified in the Storage Group Application Selection panel. The Description field is an optional field of 120 characters in which you can describe the object storage group.

To define an object storage group you must specify the following attributes:

Qualifier
This identifies a DB2 database on DASD which contains table spaces, one of which contains the object directory for this object storage hierarchy. The qualifier attribute also points to the DB2 object storage tables on DASD which have been defined for the object storage hierarchy. You can use one to eight alphanumeric characters as qualifiers.

Cycle Start Time
This is the beginning of a window of time when space management can be performed for this storage group. You must specify a value from 0 to 23, or NONE. A value from 0 to 23 represents an hour of the day. Specify an hour of the day as 00 for midnight; 01 for 1:00 a.m.; 23 for 11:00 p.m. for example. The hour of the day value for Cycle Start Time must be different from the value for Cycle End Time. Specify NONE if you do not want automatic processing for the storage group. When NONE is specified, the Cycle End Time value must be blank.

If you are running in an OAM complex and specify a cycle start time, we recommend that you also specify an OSMC processing system name, to avoid starting object processing for that object storage group on all systems in the OAM complex at cycle start time.

Cycle End Time
This is the end of a window of time when object processing can be performed for this storage group. You must specify a value of from 0 to 23, or blank (depending on what you specified for Cycle Start Time). A value of from 0 to 23 represents an hour of the day, and is required when a value from 0 to 23 was specified for Cycle Start Time. Specify an hour of the day as 00 for midnight; 01 for 1:00 a.m.; 23 for 11:00 p.m., for example. The hour of the day value for Cycle End Time must be different from the hour of the day value for Cycle Start Time. Leave the field blank if you specified NONE for Cycle Start Time.

OSMC Processing System
Specifies an OSMC processing system name, which identifies which OAM system in the sysplex performs OSMC processing for this object storage group. This prevents multiple systems from trying to process the same storage group at the same time, especially when the Cycle Start Time is used for automatically starting storage group processing.

Library Names
You can specify either one to eight real library names that represent optical libraries, or one to eight pseudo library names. The libraries you specify are used to process write requests to the storage group. Names must be valid real or pseudo optical library names defined in the SCDS.

Volume Full Threshold
This is the number of free KB triggering volume full processing for an optical volume within the object storage group. When the number of free KB falls below the threshold, the object access method marks the optical volume full. If the threshold is reached while OAM is writing an object, it continues writing that object until finished. OAM then writes no more objects to that volume. Valid values are from 0 to 9999.

Drive Start Threshold
This is the maximum number of object write requests outstanding for an optical drive in this storage group. When the number of object write requests to this storage group divided by the number of optical drives currently processing write requests for this storage group exceeds this

threshold, the object access method attempts to start an additional optical drive. Valid values are from 0 to 9999. This field is required if you specify a library name.

Volume Full at Write Error
This field indicates when to mark as “full” optical volumes within this object storage group. If Y, the object access method marks an optical volume full the first time an attempt to write an object on the optical volume fails because not enough space remains on the optical volume. If N, the object access method marks an optical volume full only when the number of available KB in the user data area falls below the Volume Full Threshold. The volume is also marked full if the optical volume table of contents area is full, regardless of how many KB are remaining in the user data area.

SMS Storage Group Status
This required field indicates whether you want to change the storage group status. If you are defining a storage group, the default value is Y. If you are altering a storage group, the default value is N and can be changed to Y or N.

You must define object storage group system statuses on the SMS Storage Group Status Define panel before the object storage group definition can be saved. After you define the status of the systems, issue the END command to save the defined object backup storage group and return to the Storage Group Application Selection panel.

Defining the Object Backup Storage Group

When you define an object backup storage group, you are naming the group that is to be used to contain backup data and, optionally, specifying a number of optical libraries that can contain backup copies of objects. There can be only one object backup storage group per system. Figure 21 on page 59 shows the Object Backup Storage Group Define panel.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                     OBJECT BACKUP STORAGE GROUP DEFINE
Command ===>

SCDS Name . . . . . : SMS.SCDS1.SCDS
Storage Group Name  : OBJB1

To DEFINE Storage Group, Specify:

  Description ==>
              ==>
  Library Names   (1 to 8 Characters each):
   ===>           ===>           ===>           ===>
   ===>           ===>           ===>           ===>

  Volume Full Threshold  . . . .       (0-9999)
  Drive Start Threshold  . . . .       (0-9999)
  Volume Full at Write Error . .       (Y or N)

   SMS Storage Group Status . . . . . . . Y  (DEFINE - Y, ALTER - Y or N)

Use ENTER to Perform Verification and Selection;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 21. Defining Object Backup Storage Group Attributes

The SCDS Name and Storage Group Name fields are output fields containing the SCDS and storage group names that you specified in the Storage Group Application Selection panel. The Description field is an optional field of 120 characters in which you can describe the object backup storage group.

To define an object backup storage group you must specify the following attributes:

Library Names
: You can specify up to eight real library names that represent optical libraries, or one to eight pseudo library names. The libraries you specify are used to process write requests to the storage group. Names must be valid real or pseudo optical library names defined in the SCDS.

Volume Full Threshold
: The number of free KB triggering volume full processing for an optical volume within the object backup storage group. When the number of free KB falls below the threshold, the object access method marks the optical volume full. If the threshold is reached while OAM is writing an object, OAM continues writing that object until finished. OAM then writes no more objects to that volume. Valid values are from 0 to 9999. This is a required field if a library name is specified.

Drive Start Threshold
: This is the maximum number of object write requests outstanding for an optical drive in this storage group. When the number of object write requests to this storage group divided by the number of optical drives currently processing requests for this storage group exceeds this threshold, the object access method attempts to start an additional optical drive. Valid values are from 0 to 9999. This is a required field if a library name is specified.

Volume Full at Write Error
: This field indicates when to mark as “full” optical volumes within this backup object storage group. If Y, the OAM marks an optical volume full the first time an attempt to write an object on the optical volume fails because not

enough space remains on the optical volume. If N, the object access method marks an optical full only when the number of available KB in the user data area falls below the Volume Full Threshold. The volume is also marked full if the optical volume table of contents area is full, regardless of how many KB are remaining in the user data area. This is a required field if a library name is specified.

SMS Storage Group Status
This required field indicates whether you want to change the storage group status. If you are defining a storage group, the default value is Y and cannot be changed to N. If you are altering a storage group, the default value is N and can be changed to Y or N.

You must define object backup storage group system statuses on the SMS Storage Group Status Define panel before the object backup storage group definition can be saved. After you define the system statuses, issue the END command to save the newly defined object storage group and return to the Storage Group Application Selection panel.

Defining Object or Object Backup Storage Group Status

After you have supplied all required fields on the Object Storage Group Define panel or the Object Backup Storage Group Define panel with valid values, press ENTER. The SMS Storage Group Status Define panel shown in Figure 22 is displayed.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                      SMS STORAGE GROUP STATUS DEFINE           Page 1 of 2
Command ===>

SCDS Name  . . . . : SMS.SCDS1.SCDS
Storage Group Name : OBJ1
Storage Group Type : OBJECT

To DEFINE Storage Group System Status, Specify:

  System           SMS SG       System         SMS SG
  Name             Status       Name           Status
 --------          ------      --------        ------
 SYSTEM1      ===> ENABLE      SYSTEM2    ===> NOTCON   ( Possible SMS SG
 SYSTEM3      ===> NOTCON      SYSTEM4    ===> NOTCON     status value for
 SYSTEM5      ===> NOTCON      SYSTEM6    ===> NOTCON     each System: NOTCON,
              ===>                        ===>            ENABLE, DISALL,
              ===>                        ===>            or DISNEW )
              ===>                        ===>
              ===>                        ===>          ( Only system names
              ===>                        ===>            are displayed )
Use ENTER to Perform Verification; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 22. Defining Object or Object Backup Storage Group System Status

Notice that only systems and not system groups are shown on this panel. This is because you cannot connect an object or object backup storage group to a system group.

The SCDS Name, Storage Group Name and Storage Group Type fields are output fields that contain the information you specified on the Storage Group Application Selection panel. The System Name field is also an output field. It lists the systems that you defined in the base configuration. The SMS Storage Group Status field lists the relationship between the object or object backup storage group and each system in the SMS complex.

| The devices associated with an object or object backup storage group have both physical (actual) and logical (system defined) connections to systems in the SMS complex. You are responsible for maintaining consistency between these two types of connections.

One of the following four relationships exists between each system in the SMS complex and the object or object backup storage group:

ENABLE
For object storage groups, SMS permits application access to the object storage hierarchy of this group. For object backup storage groups, SMS permits access to the volume set. All OSREQ functions are allowed.

DISALL
For object or object backup storage groups, SMS permits restricted access to the object storage hierarchy. The OSREQ functions STORE, RETRIEVE, and DELETE are denied to applications. All object processing continues to be done for the storage group.

DISNEW
For object or object backup storage groups, SMS permits restricted access to the hierarchy. The OSREQ function STORE is denied to applications. All object processing continues to be done for the storage group.

NOTCON
The named system cannot process this object/object backup group.

NOTCON is the default status.

You can enable object and object backup storage groups to more than one system. If you are not in an OAM complex, OAM ignores object storage groups that are defined as being connected to more than one system and issues a message.

After establishing the relationships between the object or object backup storage group and the systems in the SMS complex, you need to complete the definition of the storage group. From the SMS Storage Group Status Define panel, enter the END command to save your values and return to the Object Storage Group or Object Backup Storage Group Define panel. On this panel, enter the END command to save the storage group.

Assigning an OAM Object Collection to a Storage Group

OAM object collections are always managed by SMS. The ACS routines are executed when the OAM object collection catalog entry is initially defined. The ACS routines provide storage class and management class names that are used as the default storage class and management class assignments recorded in the catalog entry, and the storage group assignment for the collection. OAM uses the storage group assignment to identify the DB2 tables used for storage of the object directory information, for storage of objects on DASD, and to select the optical volume used to store objects that belong to the collection.

Defining a Tape Storage Group

Figure 23 shows the Tape Storage Group Define panel.

The SCDS Name and Storage Group Name fields are output fields containing the

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                         TAPE STORAGE GROUP DEFINE
Command ===>

SCDS Name . . . . . : SMS.SCDS1.SCDS
Storage Group Name  : TAPE1

To DEFINE Storage Group, Specify:

  Description ==>
              ==>

  Library Names  (1 to 8 characters each):
    ===>           ===>           ===>           ===>
    ===>           ===>           ===>           ===>



  DEFINE SMS Storage Group Status . .... N  (Y or N)


Use ENTER to Perform Verification and Selection;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 23. Defining Tape Storage Group Attributes

SCDS and storage group names that you specified in the Storage Group Application Selection panel. The Description field is an optional field of 120 characters in which you can describe the tape storage group. ISMF primes the field with blanks, which are the default. Issue the END command to save the newly defined tape storage group and return to the Storage Group Application Selection panel.

The Library Names field is used to specify the tape libraries that own the volumes within this storage group. One to eight library names can be associated with a tape storage group. At least one library name must be specified when defining a tape storage group. The library name in the tape storage group definition must also be defined in the same SCDS.

The SMS Storage Group Status field is used to specify whether or not you want to modify the status (enable, disall, disnew, notcon) of this storage group for each one of the systems in the SMS complex. If Y is specified, the SMS Storage Group Status Define panel is displayed. If N is specified, the status panel is not displayed, and the storage group defaults to ENABLE for each one of the systems in the complex. This is a required field. N is the default.

Defining Tape Storage Group Status

In the Define SMS Storage Group Status field, indicate whether you want to change the status of the tape storage group with respect to a given system in the SMS complex. Initially, all of the status fields are ENABLE. Specify Y, yes, to change a status. Accept the default N, no, to leave all the status fields as ENABLE.

When you have completed all your entries in the Tape Storage Group Define panel, press ENTER. If you specified Y for the Define SMS Storage Group Status attribute, you get the SMS Storage Group Status Define panel shown in Figure 24 , where you can define the status of the storage group to each system in the SMS complex.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                      SMS STORAGE GROUP STATUS DEFINE          Page 1 of 2
Command ===>

SCDS Name  . . . . : SMS.SCDS1.SCDS
Storage Group Name : TAPE1
Storage Group Type : TAPE
To DEFINE Storage Group System/                              ( Possible SMS SG
Sys Group Status, Specify:                                  Status for each:
                                                            - Pool SG Type
 System/Sys      SMS SG    System/Sys      SMS SG             NOTCON, ENABLE
 Group Name      Status    Group Name      Status             DISALL, DISNEW
 ----------      ------    ----------      ------             QUIALL, QUINEW
 *SYSPLX1   ===> ENABLE    SYSTEM2    ===> ENABLE       - Tape SG Type
 SYSTEM3    ===> NOTCON    SYSTEM4    ===> NOTCON         NOTCON, ENABLE,
            ===>                      ===>                DISALL, DISNEW )
            ===>                      ===>
            ===>                      ===>              * SYS GROUP = sysplex
            ===>                      ===>                minus Systems in the
            ===>                      ===>                Sysplex explicitly
            ===>                      ===>                defined in the SCDS
Use ENTER to Perform Verification; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 24. Defining Tape Storage Group System Status

The SCDS Name, Storage Group Name and Storage Group Type fields are output fields that contain the information you specified on the Storage Group Application Selection panel. The System/Sys Group Name field is also an output field. It lists the systems that you defined in the base configuration (using the CDS application). The SMS Storage Group Status field lists the relationship between the storage group and each system in the SMS complex. See “Defining System Access to Pool and VIO Storage Groups” on page 47 for an explanation of these relationships.

From the SMS Storage Group Status Define panel, enter the END command to save your values and return to the Tape Storage Group Define panel. On this panel, enter the END command to save the tape storage group. This returns you to the Storage Group Application Selection panel, which is primed with your CDS and storage group name.

Defining Additional Storage Groups

You can copy existing storage groups and modify them to create new storage groups by using the COPY line operator, which is explained in “Copying SMS Classes, Storage Groups, and Aggregate Groups” on page 204.

When you copy either a pool or VIO storage group within an SCDS, all of the system status values and system names for automatic processing are copied. When you copy either a pool or VIO storage group from one SCDS to a different SCDS, all of the system status values default to ENABLE rather than to the system status values of the source storage group. Also, the Migrate System Name, Backup System Name, and Dump System Name fields are set to blank.

Whenever you copy an object storage group within an SCDS, you must modify the qualifier to ensure that the new object storage group has a unique qualifier for the connected system. All qualifiers within an system must be unique. Specify Y for the Perform Alter attribute on the Copy Entry panel to display the Object Storage Group Alter panel. Modify the qualifier and issue an END command to save the new qualifier.

Whenever you copy an object storage group from one SCDS into a different SCDS, the qualifier must be unique within the target SCDS. If necessary, alter the qualifier in the same manner as required for copying an object storage group within an SCDS.

The system status values of object and object backup storage groups are retained when copying within an SCDS. Object and object backup storage groups can be enabled by and connected to more than one system. Specify Y for the Perform Alter attribute on the Copy Entry panel, and Y for the Alter SMS Storage Group Status attribute on either the Object Storage Group Alter panel or Object Backup Storage Group Alter panel to display the system statuses for modification.

Listing Volumes in a Storage Group

Select option 1, List, on the Storage Group Application Selection panel, Figure 13 on page 45, to get a listing of the volumes in the storage group specified in the panel. Enter the LISTVOL line operator next to the storage group to select a list. The LISTVOL line operator allows you to list volume information, such as volume serial numbers, for a pool, dummy, object, object backup storage group or tape storage group. You can use the LISTVOL storage group line operator without any parameters to list all volumes in a specified storage group. See the example shown in Table 1. The list displayed depends on the storage group type.

The DASD Volume STATUS line operator allows users to display 32 sets of SMS and MVS volume statuses corresponding to 32 system names, system group names or both for a volume from the Volume List panel. The Volume List panel only displays eight of the possible 32 sets of SMS and MVS volume statuses at a time. In order to view volume statuses beyond the first eight sets, this line operator has to be entered against the volume on the Volume List Panel. This line operator STATUS can be abbreviated to a minimum of the first two characters ″ST″.

Table 1. Volumes Listed in a Specified Storage Group

Storage Group Type	Volume List Shown
POOL DUMMY	DASD Volume List
OBJECT or OBJECT BACKUP	Mountable Optical Volume List
TAPE	Mountable Tape Volume List

Please refer to *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support* for information on the Mountable Optical Volume List, and to *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Tape Libraries* for information on the Mountable Tape Volume List.

Chapter 5. Defining Management Classes

DFSMShsm manages non-SMS data sets at the volume level, applying the management criteria to all data sets on a given volume. SMS automates the management of storage at the data set level by introducing management classes. When assigned to data sets, management classes replace and expand attributes that otherwise would be specified on JCL DD statements, IDCAMS DEFINE commands, and DFSMShsm commands. When assigned to objects, OAM uses a subset of the management class attributes and the OSMC class transition attributes to manage objects. This chapter describes management classes and shows you how to define them using the ISMF management class application.

With the introduction of the Copy Technique attribute, the storage administrator has the option of specifying whether a system-managed backup process called concurrent copy should be used for data sets to enhance system availability during data set backup and aggregate backup processing. Data Set Alter can be used to alter a management class so data sets can use the concurrent copy technique during backup processing. The storage administrator has the option of specifying whether the concurrent copy process is required, preferred, or discouraged. The Copy Technique attribute is related to the Backup and Aggregate Backup components of management class. A Copy Technique field exists for each of the Backup and Aggregate Backup components. These fields appear on the management class list, display, define/alter, view, and sort panels.

Understanding Management Classes

A management class is a list of data set migration, backup and retention attribute values. Management class also includes object expiration criteria, object backup requirements, and class transition criteria for management of objects. DFSMShsm uses the attributes of the management class associated with a data set to manage storage. When you assign a management class to a system-managed DASD data set, SMS places the management class name in both the basic catalog structure (BCS) and the VSAM volume data sets (VVDS) catalog entries of the data set. Management class is optional for system-managed data sets and does not apply to non-system-managed data sets. Management class is *not* used for tape datasets.

If you alter a management class definition, SMS applies the changes to any new data sets or objects after you activate the changed configuration. SMS also applies the changes to any previously allocated data sets or objects, beginning with their next scheduled management cycle (such as daily space management or backup).

Default Management Class

For data sets, you can specify a default management class in an SCDS. DFSMShsm applies the default management class to all system-managed data sets that do not already have a management class. Unlike the data sets that you have already assigned a management class, the catalog entries of these data sets do not contain a management class name. For objects, the default management class is defined in the catalog entry for the object collection to which the objects belong. The default management class is assigned by the management class ACS routine when the first object is stored in the collection.

OAM Management Classes

OAM uses some attributes in the management class associated with the object to manage the object. Class transition attributes allow OAM to change the way an object is managed based on its age, its usage, or a predefined periodic function. A class transition is a change in the object's management class or storage class when an event occurs which brings about a change in an object's management criteria or service level. Class transitions occur during an OSMC storage management cycle. Objects requiring class transition use the ACS routines to determine if they should be managed using a different management class or placed at a different level of the storage hierarchy according to a new storage class.

OAM uses the backup attributes of the management class definition to initiate the writing of an optical backup copy of an object. This copy might be made during the first storage management cycle after the object has been stored and a management class with backup attributes has been assigned to the object.

Retention attributes determine the OAM action for object expiration. An object can expire automatically based on its age, its usage, or a specific date derived from its management class and an object-specific retention period, if provided. OSMC deletes expired objects from the directory automatically during the storage management cycle with the approval of the auto-delete installation exit.

| For a more specific discussion of how to use management classes with objects, see
| *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for*
| *Object Support*.

Describing Management Classes

A management class definition contains both descriptive and storage-related information. To identify and refer to your management classes, you assign each one a unique name that contains from one to eight alphanumeric characters. Each management class definition maintains an owner ID that identifies the storage administrator who originally created or last modified the management class. The *owner ID* is an MVS user ID on the ISMF Management Class List in column 19—Last Modified Userid. Also, each management class contains an optional 120 character description field for describing its contents.

Planning Management Classes

You should create a management class for each type of service that is to be provided by the installation. A type of service is defined for a collection of data sets that have similar migration, backup, and retention requirements or objects that have similar backup, expiration, and class transition requirements. Before you actually define your management classes, you should gather the following information:

- Requirements for releasing overallocated space
- Migration requirements
- Retention criteria
- Treatment of expired data sets
- Frequency of backup
- Number of backup versions
- Retention of backup versions
- Number versions

- Retain only version
- Retain only version unit
- Retain extra versions
- Retain extra versions unit
- Copy serialization
- Generation data group (GDG) information
- Object class transition criteria

Based on this information, you can establish management classes that centralize storage management in the SMS complex.

If a data set has a management class that specifies automatic backup or migration, then you must direct the data set to storage groups that are eligible to be processed for automatic backup or migration capabilities.

A generation data group (GDG) is a group of related cataloged data sets that have sequentially ordered names. SMS uses GDG-related information in the management class definition to manage the storage associated with these data sets. Some management criteria is specified in the definition of the GDG. Some management criteria might be specified by assigning a management class to each individual generation in the GDG. Generation data sets within the same GDG might have different management classes assigned by JCL.

Defining Management Class Attributes

You can use ISMF to define your management classes by selecting option 3, Management Class, from the ISMF Primary Option Menu for storage administrators. Figure 25 on page 68 and Figure 26 on page 69 illustrate the Management Class Application Selection panels.

Note: For additional information on option 1, List, see "Listing SMS Classes, Aggregate Groups, Storage Groups, and Libraries Using ISMF" on page 190. See *DFSMS/MVS Using ISMF* for more information on option 2, DISPLAY, and see "Altering Management Classes" on page 197 for more information on option 4, Alter.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                    MANAGEMENT CLASS APPLICATION SELECTION         Page 1 of 2
Command ===>

To perform Management Class Operations, Specify:
  CDS Name  . . . . . . . . . 'SMS.SCDS1.SCDS'
                               (1 to 44 character data set name or 'Active' )
  Management Class Name . . . INTERIM   (For Management Class List, fully or
                                        partially specified or * for all)

Select one of the following options  :
  3  1. List    - Generate a list of Management Classes
     2. Display - Display a Management Class
     3. Define  - Define a Management Class
     4. Alter   - Alter a Management Class:

If List Option is chosen,
   Enter "/" to select option      Respecify View Criteria
                                   Respecify Sort Criteria


Use ENTER to Perform Selection; Use DOWN Command to View next Selection Panel;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 25. Management Class Application Selection Panel for Storage Administrators, Page 1 of 2

To define a management class, you must specify a CDS Name and a Management Class Name on the panel and select option 3, Define. The CDS Name must be the name of an SCDS. ISMF primes the field with the last used name. The default is the quoted word 'ACTIVE', which represents the currently active configuration, but you cannot define or alter management classes to the 'ACTIVE' configuration.

In the Management Class Name field, you must specify the name of the management class that you are defining. Your management class names should be generic rather than specific, so that attribute changes do not make names meaningless or misleading. STANDARD and INTERIM are examples of generic names. RETAIN30 and BUDAILY are examples of specific names. ISMF primes the field with the name last used.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                    MANAGEMENT CLASS APPLICATION SELECTION         Page 2 of 2
Command ===>

CDS Name  . . . . . . . : SMS.SCDS1.SCDS
Management Class Name . : INTERIM


If option other than List is chosen in the previous page, Select attribute
group to Display, Define or Alter first :

   1  1. Space Management Attributes  -  Pages 1 and 2
      2. Backup Attributes            -  Page  3
      3. Class Transition Attributes  -  Page  4
      4. Aggregate Backup Attributes  -  Page  5






Use ENTER to Perform Selection; Use UP Command for previous Selection Panel;
Use HELP Command for Help; Use END Command to Exit;
```

Figure 26. Management Class Application Selection Panel for Storage Administrators, Page 2 of 2

Designation of an attribute group is only useful when performing a display, define, or alter function. An attribute group is selected for processing ahead of other Management Class attributes by entering its corresponding number in the selection field provided. Panels containing the other management class attributes can be accessed by paging up and down in the usual manner.

Defining Management Class Expiration Attributes

Figure 27 on page 70 shows the first page of the Management Class Define panel. You can leave any page of the Management Class Define panel without saving the management class by issuing the CANCEL command.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                         MANAGEMENT CLASS DEFINE              Page 1 of 5
Command ===>

SCDS Name . . . . . . : SMS.SCDS1.SCDS
Management Class Name : INTERIM

To DEFINE Management Class, Specify:

  Description ==>
              ==>

  Expiration Attributes
    Expire after Days Non-usage  . . 2           (1 to 9999 or NOLIMIT)
    Expire after Date/Days . . . . . 7           (0 to 9999, yyyy/mm/dd or
                                                 NOLIMIT)

  Retention Limit  . . . . . . . . . NOLIMIT     (0 to 9999 or NOLIMIT)



Use ENTER to Perform Verification; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 27. Defining Management Class Expiration Attributes

Page one contains the management class expiration attributes. DFSMShsm processes the expiration attributes before the migration attributes that you specify on the second page of the Management Class Define panel. SCDS Name and Management Class Name are output fields that contain the SCDS and management class names you specified in the Management Class Application Selection panel. Description is an optional field of 120 characters where you can describe the management class.

Expiration Attributes and Retention Limit

You use expiration attributes to determine the action for data set and object expiration and deletion. DFSMShsm deletes expired data sets during automatic space management processing. OAM deletes objects during a storage management cycle with the approval of the auto-delete installation exit. Expiration attributes are required values that indicate when a data set or object becomes eligible for expiration. You can base expiration criteria on a specific date, on the number of days since the data set or object was allocated, or on the number of days since the data set or object was last referenced.

The Expire after Days Non-usage field specifies how much time must elapse since last access before a data set or object becomes eligible for expiration. The Expire after Date/Days field specifies an absolute date or period after its allocation for a data set or object to become eligible for expiration. The default for both fields is NOLIMIT.

The Retention Limit value is a required value that limits the use of retention period (RETPD) and expiration date (EXPDT) values that are explicitly specified in JCL, are derived from data class definitions or are explicitly specified in the OSREQ STORE macro. If the value of a user-specified RETPD or EXPDT is within the limits specified in the Retention Limit field, it is saved for the data set. For objects, only RETPD is saved.

The default retention limit is NOLIMIT. If you specify zero, a user-specified or data class derived EXPDT or RETPD is ignored. If users specify values that exceed the maximum period, the retention limit value overrides not only their values but also the expiration attributes values. The retention limit value is saved. ISMF primes the Retention Limit field with what you specified the last time.

DFSMShsm determines if a data set has expired based on the expiration date found in the catalog entry of the data set. OAM determines if an object has expired based on the expiration date in its object directory entry. If an expiration date is not found, DFSMShsm and OAM use the management class expiration attributes. These attributes are used as follows:

- If both expiration attributes are NOLIMIT, the data set or object never expires.
- If one of the expiration attributes is NOLIMIT, then the other attribute must be satisfied.
- If neither expiration attribute is NOLIMIT, both of the expiration attributes must be satisfied.

Note: If you want to change the expiration date of a catalog entry, use the access method services ALTER command. For information on changing the expiration date of an object, see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support*.

Data sets or objects having the INTERIM management class defined in Figure 27 on page 70 become eligible for expiration when both of the following criteria are met; at least seven days since allocation and not referenced in the last two days.

Table 2 shows several combinations of retention attributes used for space management processing. Use the highlighted values for each instance.

Table 2. Comparing Retention Period Attributes

Attribute	Case 1	Case 2	Case 3	Case 4	Case 5
Retention Limit	0	**50**	100	NOLIMIT	100
Expire after Days Non-usage	**50**	50	50	50	50
Expire after Date/Days	**100**	100	100	100	**100**
RETPD/EXPDT	60	60	**60**	**60**	

In the first case, the retention limit is zero, so DFSMShsm and OAM honor the values of 50 and 100 which are the values specified for Expire after Days Non-usage and Expire after Date/Days and ignore any user-specified or data class values. In Case 2, the management class expiration values are ignored because RETPD and EXPDT values have been specified or derived and the retention limit is nonzero. However, because the retention limit is less than the user-specified or data class values, 50 is saved and used to calculate the expiration date.

In Case 3, the user-specified or data class values fall within the retention limit. So, DFSMShsm uses the values of 60 and 60. OAM uses a RETPD value of 60, because OAM does not consider EXPDT.

In Case 4, the RETPD and EXPDT are used because the retention period value is NOLIMIT.

In Case 5, because no user-specified or data class derived values are available, DFSMShsm and OAM use the values specified in the management class expiration attributes which are shown on the Management Class Define panel.

After you specify the expiration attributes, issue the DOWN command to see the next page of the Management Class Define panel, which is shown in Figure 28, on which you can specify the migration attributes.

Defining Management Class Migration Attributes

To DFSMShsm, a data set occupies one of two distinct states in storage:

Primary
: Also known as *level 0*, the primary state indicates that end users can directly access a data set residing on a volume.

Migrated
: End users cannot directly access data sets that have migrated from the primary state to a migrated state. To be accessed, the data sets must be recalled to primary storage. A migrated data set can reside on either *migration level 1* (usually permanently mounted DASD) or *migration level 2* (usually tape).

A data set can move back and forth between these two states, and it can move from level 0 to migration level 2 (and back) without passing through migration level 1. Objects do not migrate. Movement back to level 0 is called *recall*.

Figure 28 shows the second page of the Management Class Define panel, which contains the management class migration and GDG management attributes. SCDS Name and Management Class Name are output fields that contain the SCDS and management class names you specified in the Management Class Application Selection panel.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                         MANAGEMENT CLASS DEFINE               Page 2 of 5
Command ===>

SCDS Name . . . . . . : SMS.SCD1.SCDS
Management Class Name : INTERIM

To DEFINE Management Class, Specify:

  Partial Release . . . . . . . . . Y           (Y, C, YI, CI or N)

  Migration Attributes
    Primary Days Non-usage  . . . .             (0 to 9999 or blank)
    Level 1 Days Non-usage  . . . .             (0 to 9999, NOLIMIT or blank)
    Command or Auto Migrate . . . . NONE        (BOTH, COMMAND or NONE)


  GDG Management Attributes
    # GDG Elements on Primary . . . 1           (0 to 255 or blank)
    Rolled-off GDS Action . . . . . EXPIRE      (MIGRATE, EXPIRE or blank)

Use ENTER to Perform Verification; Use UP/DOWN Command to View other Panels;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 28. Defining Management Class Migration Attributes

To define a management class, you must specify values for the Partial Release and Command or Auto Migrate attributes. If you specify BOTH for the Command or Auto Migrate attribute, then Primary Days Non-usage and Level 1 Days Non-usage become required fields. If you specify NONE for the Command or Auto Migrate attribute, the expiration attributes specified on page 1 of this panel still apply. The other fields on this panel are optional.

Partial Release Attribute

The Partial Release attribute applies to non-VSAM data sets and to all VSAM data set types allocated in the extended format. You can use partial release to choose the conditions under which unused allocated space is released. If you select partial release, space release is carried out automatically at the time you have selected. Each partial release option releases the same amount of space. Unused space is released in cylinders or tracks, depending on the space allocation unit used. The following options are available for partial release:

Y Yes. Release unused space at Space Management cycle time.

YI Yes Immediate. Release unused space at Space Management cycle time and also when close is issued for a data set that was open for output.

C Conditional. If a nonzero secondary space allocation has been specified for the data set, release unused space at Space Management cycle time. This option only applies to physical sequential and partitioned data sets. If specified for an extended format VSAM data set, this option is processed as if Yes were specified.

CI Conditional Immediate. If secondary space has been allocated, release unused space at Space Management cycle time and also when close is issued for a data set that was open for output.

N No. No release of unused space.

Note:

ISMF primes the field with the last used value. The default is N.

Notes:

1. The Partial Release attribute is not checked during DFSMShsm interval migration.
2. The Partial Release attribute is mutually exclusive with the Guaranteed Space storage class attribute.
3. For more information on partial release, see *DFSMS/MVS Using Data Sets*.
4. Partial release is ignored under the following conditions:
   - Another job is sharing the data set (DISP=SHR)
   - Another task in the same job is processing an OPEN, CLOSE, EOV, or FEOV request for the data set
   - Another DCB is open for the data set.

Migration Attributes

The Primary Days Non-Usage attribute represents the minimum number of days that must elapse since last access before a data set is eligible for normal migration. Days refers to calendar days, not to a 24-hour time period. In other words, the day you close the data set counts as the first full day of non-usage. The minimum number of Primary Days Non-usage days required for migration to ML2 includes the time already spent unreferenced on ML0. The default is 2. (See “GDG Management Attributes” for the special GDS migration eligibility rules.)

A nonzero number for Level 1 Days Non-usage indicates the number of days that must elapse since the last reference on level 0 before a data set on ML1 can migrate from level 1 to level 2. The default is 60.

The Command of Auto Migrate attribute allows you to specify if a data set is eligible to be migrated by BOTH command and automatic processing, by command alone, or not at all (NONE).

If you do not want to migrate data sets that belong to a particular management class, specify NONE in the Command or Auto Migrate field. The data sets remain on primary storage until they expire. If you want data sets to be eligible for migration directly from level 0 storage to level 2 tape, specify 0 in the Level 1 Days Non-usage field. Otherwise, the data sets must first migrate to level 1 (the days spent on level 0 count toward the eligibility of moving to level 2).

GDG Management Attributes

The GDG Management Attributes indicate criteria for early migration of GDGs off primary storage and what to do with rolled-off generation data sets (GDSs).

The # GDG Elements on Primary indicates how many of the most recent generations of a GDG use the normal Primary Days Non-usage attribute for migration criteria. Generations that are over this limit are eligible for early migration during the next processing of primary space management. The inactive age is irrelevant for these ’over the limit’ GDSs and they are given a higher selection priority so they are likely to be migrated if space is needed to meet volume thresholds.

Note: The # GDG Elements on Primary attribute does not ensure that the specified number of GDSs are kept on primary storage. It is just that their migration is not accelerated.

To specify that old generations are to have priority for migration and to specify how many generations are to have standard priority, you can specify the # GDG Elements on Primary attribute. When you specify this attribute, all non-rolled-off generations of each GDG that exceed the number you specify are given priority for early migration. If you specify 0 for this attribute, all generations of the GDGs are given priority for early migration. Those data sets made eligible for early migration do not have to satisfy the Primary Days Non-usage criteria.

All generations that meet the Primary Days Non-usage criteria are eligible to migrate. # GDG Elements on Primary enables you to accelerate migration of older generations while Primary Days Non-usage enables you to specify migration criteria of younger generations separately.

The Rolled-off GDS Action value indicates whether to expire rolled-off GDSs or to make them eligible for migration. If you specify MIGRATE, management class expiration attributes are applied to the data set to determine if the data set should be deleted; if not, the data set is eligible for migration. If you specify EXPIRE, the rolled-off GDS is deleted unless an explicit expiration date is in the DSCB. In this case, the rolled off GDS becomes a non-GDG data set and migrates according to non-GDG migration attributes.

Note: The DFSMShsm parameter, EXPIREDATASET(SCR | NOSCR), controls the deletion of a data set with an expiration date in the DSCB.

Both the # GDG Elements on Primary and the Rolled-off GDS Action attributes are optional and have default values of blank. If left blank, no special treatment results from the data set being a GDS or a rolled-off GDS. That is, other management class attributes is used to process the data set.

During automatic primary space management, GDSs that have been determined as eligible for migration are migrated in the following priority order:

1. GDSs that are rolled-off.
2. GDSs that exceed the limit specified in # GDG Elements on Primary.
3. Other GDSs are handled by the non-GDG algorithm which bases priority on a function of size and the length of time it has been eligible to migrate in Primary Days Non-usage.

After specifying the migration attributes, issue the DOWN command to specify the backup attributes on the next page of the Management Class Define panel, which is shown in Figure 29.

Defining Management Class Backup Attributes

The third page of the Management Class Define panel contains the management class backup attributes. (For objects, see "Defining Object Class Backup Attributes" on page 78.)

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                         MANAGEMENT CLASS DEFINE            Page 3 of 5
Command ===>

SCDS Name . . . . . . : SMS.SCDS1.SCDS
Management Class Name : INTERIM

To DEFINE Management Class, Specify:
  Backup Attributes
    Backup Frequency  . . . . . . . . 1        (0 to 9999 or blank)
    Number of Backup Vers . . . . . . 2        (1 to 100 or blank)
      (Data Set Exists)
    Number of Backup Vers . . . . . . 1        (0 to 100 or blank)
      (Data Set Deleted)
    Retain days only Backup Ver . . . 60       (1 to 9999, NOLIMIT or blank)
      (Data Set Deleted)
    Retain days extra Backup Vers . . 30       (1 to 9999, NOLIMIT or blank)
    Admin or User command Backup  . . NONE     (BOTH, ADMIN or NONE)
    Auto Backup . . . . . . . . . . . Y        (Y or N)
    Backup Copy Technique . . . . . . S        (P=Conc Preferred, R=Conc
                                               Required or S=Standard)
Use ENTER to Perform Verification; Use UP/DOWN Command to View other Panels;
Use HELP Command for Help; Use END Command to Save and Exit; Cancel to Exit.
```

Figure 29. Defining Management Class Backup Attributes

Backup Frequency

The Backup Frequency attribute specifies how many days must elapse before DFSMShsm can back up data sets that have changed since the last backup. If you want to back up changed data sets every time that DFSMShsm processes the volumes containing them, specify 0. The default is 1.

This field does not apply to objects.

Number of Backup Versions

The Number of Backup Versions fields specify the maximum number of backup versions to retain for a data set. The default is 2 if the data set still exists and 1 if it has been deleted.

For objects, only one backup copy is made. When the original object is deleted, the backup copy is also deleted.

Retain Days Only Backup Versions

The Retain days only Backup Version (Data Set Deleted) field indicates how many days to keep the most recent backup version of a deleted data set, starting from the day DFSMShsm detects it has been deleted. This attribute applies only when a data set no longer exists on primary (level 0) or migrated (levels 1 and 2) storage. The default is 60.

This field does not apply to objects. Backup copies of objects are not retained when the original object is deleted.

Retain Days Extra Backup Versions

The Retain days extra Backup versions field indicates how many days to keep backup versions other than the most recent one, starting from the day backups were created. It only applies when more than one backup version exists, and when a data set has low activity. When a new backup version is created and the number of backup versions already equals the value specified in the pertinent (existing or deleted) Number of Backup Versions field, the oldest version is deleted. This attribute applies whether the data set has been deleted or not. The default is 30.

This field does not apply to objects. An object can have only one backup copy.

Admin or User command Backup

The Admin or User command Backup field indicates if both the end user and the storage administrator can issue command backups of the data sets in this management class, if only the storage administrator can, or if neither of them can. The Admin or User command Backup field is required, and has a default value of BOTH. Otherwise, the remaining fields are optional.

BOTH indicates that both storage administrators and users can perform command backups against these data sets.

ADMIN
indicates that only storage administrators can perform command backups against these data sets.

NONE indicates that neither storage administrators or end users can perform command backups.

If you specify BOTH or ADMIN in the Admin or User Command Backup field, the remaining fields on this panel are required. If the value is BOTH or ADMIN, then AUTO BACKUP specifies whether these data sets are eligible for automatic backup.

Auto Backup

The Auto Backup field is required, and has a default value of Y. If you specify Y in the Auto Backup field, the remaining fields on this panel are required. Otherwise, the remaining fields are optional.

The only field used by OAM for objects is Auto Backup. OAM ignores all fields except Auto Backup. Only one backup copy of an object is made, since objects cannot be changed. The object is backed up during the first management cycle after the object is assigned to a management class requiring backup. This can occur when the object is stored or when its management class is changed by you or by a class transition. The backup copy of an object is deleted when the object is deleted. No archived copy is saved.

Backup Copy Technique

The Backup Copy Technique field specifies whether concurrent copy should be used during data set backup processing.

Note: When deciding whether or not to use concurrent copy for a particular dump or copy operation, you need to decide why you are making a copy of the data. If the intention of the copy is to capture a ″point-in-time″ image of the data at a specific time, concurrent copy might not be appropriate. When you use concurrent copy there is always a chance that a dump or copy operation does not complete once you have started to make updates to the data. If this happens, the specific ″point-in-time″ is lost and you might not be able to recover an image of the data at that time.

For more information see *International Technical Support Centers Implementing Concurrent Copy*

R indicates that concurrent copy must be used for backup. Backup fails for data sets that don't reside on volumes supported by concurrent copy, or that are unavailable for concurrent copy.

P indicates that concurrent copy is preferred and should be used for backup. A data set is backed up on a non-concurrent copy volume if it does not reside on a volume supported by concurrent copy, or if the volume on which it resides is unavailable for concurrent copy.

S indicates standard allocation, in which data sets are backed up without using concurrent copy.

Note that you should take special care to avoid specifying conflicting conditions in the storage class and the management class. For example, if you specify Backup Copy Technique as R (required) in the management class and Accessibility as NOPREF in the storage class, you would be requesting concurrent copy for backup while having placed the data set on a device that does not support concurrent copy, which would result in the backup failing. Specifying the backup copy technique option in the management class in no way affects the placement of data sets on a concurrent copy volume.

After you specify the backup attributes, issue the DOWN command to see the next page of the Management Class Define panel, which is shown in Figure 30 on page 78, on which you can specify the object attributes.

Defining Object Class Backup Attributes

Page 4 of the Management Class Define panel, shown in Figure 30, contains the object class transition attributes.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                         MANAGEMENT CLASS DEFINE              Page 4 of 5
Command ===>

SCDS Name . . . . . . : SMS.SCDS1.SCDS
Management Class Name : INTERIM

To DEFINE Management Class, Specify:
  Object Class Transition Criteria
    Time Since Creation Years  . .       (0 to 9999 or blank)
                        Months . .       (0 to 9999 or blank)
                        Days . . .       (0 to 9999 or blank)
    Time Since Last Use Years  . .       (0 to 9999 or blank)
                        Months . .       (0 to 9999 or blank)
                        Days . . .       (0 to 9999 or blank)
    Periodic
      Monthly   On Day . . .        (1 to 31, FIRST, LAST or blank)
      Quarterly On Day . . .        (1 to 92, FIRST, LAST or blank)
                In Month . .        (1 to 3 or blank)
      Yearly    On Day . . .        (1 to 366, FIRST, LAST or blank)
                In Month . .        (1 to 12 or blank)
Use ENTER to Perform Verification; Use UP/DOWN Command to View Other Panels;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 30. Defining Object Class Backup Attributes

The Time Since Creation Years, Months, or Days fields indicate the time since the creation date that must pass before transition occurs.

The Time Since Last Use Years, Months, or Days fields indicate the time since the last reference date that must pass before transition occurs.

The Periodic field indicates a time based on the calendar at which transition occurs.

Note: The Time Since Creation, Time Since Last Use and Periodic fields cannot be specified together. A maximum date of 9999/12/31 is used if the requested Time Since Creation or Time Since Last Used exceeds the maximum date.

The Monthly On Day field specifies the day of each month that the transition occurs. If there are fewer days in the month than the number specified, the transition occurs on the last day of the month.

The Quarterly On Day or In Month fields specify the time of each quarter that the transition occurs. If both Day and Month are specified, this attribute specifies the day of the month in each quarter that the transition occurs. If there are fewer days in the specified month than the number specified in Day, then the transition occurs on the last day of the specified month.

The Yearly On Day or In Month fields specify the day or month of each year that transition occurs. If both Day and Month are specified, this attribute specifies the day of the month in each year that transition occurs.

FIRST specifies that the transition occurs on the first day of each month, quarter, or year, whichever attribute is specified.

LAST specifies that the transition occurs on the last day of each month, quarter, or year.

Defining Aggregate Backup Attributes

The fifth page of the Management Class Define panel shown in Figure 31 contains the aggregate backup attributes.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                         MANAGEMENT CLASS DEFINE              Page 5 of 5
Command ===>

SCDS Name . . . . . . : SMS.SCDS1.SCDS
Management Class Name : INTERIM

To DEFINE Management Class, Specify:
  AGGREGATE Backup Attributes:
    # Versions . . . . . . . .         (1 to 9999, NOLIMIT or blank)
    Retain Only Version  . . .         (1 to 9999, NOLIMIT or blank)
      Unit . . . . . . . . . .         (D=days, W=weeks, M=months, Y=years or
                                       blank)
    Retain Extra Versions  . .         (1 to 9999, NOLIMIT or blank)
      Unit . . . . . . . . . .         (D=days, W=weeks, M=months, Y=years or
                                       blank)
    Copy Serialization . . . .         (C=continue, F=fail or blank)
    Abackup Copy Technique . . S       (P=Conc Preferred, R=Conc Required or
                                       S=Standard)


Use ENTER to Perform Verification; Use UP Command to View previous Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 31. Defining Aggregate Backup Attributes

The SCDS Name field specifies the name of the SCDS into which the management class is defined.

The # Versions field specifies the number of versions to be maintained.

The Retain Only Version field indicates how long the most recent backup version of an aggregate group is kept.

The Unit field specifies the unit of measure for the time period specified for the Retain Only Version field. This field cannot be blank if the Retain Only Version field is specified, except when Retain Only Version is NOLIMIT.

The Retain Extra Versions field indicates how long to keep backup versions of an aggregate group that precede the most recent version.

The Unit field specifies the unit of measure for the time period specified for the Retain Extra Versions field. This field cannot be blank if the Retain Extra Versions field is specified, except when Retain Extra Versions is NOLIMIT.

The Copy Serialization field specifies whether aggregate backup should continue if an enqueue failure is encountered.

The Abackup Copy Technique field specifies whether concurrent copy should be used in conjunction with aggregate group backup.

Assigning Management Classes

You can assign management classes through the management class ACS routine, by explicit specification or as part of an object class transition. For permanent system-managed data sets, the management class ACS routine is executed only if the storage class is valid. The management class ACS routine is not executed for temporary data sets.

An object's management class is determined when you store the object. The default management class for an object is defined in the collection to which the object belongs. (The default management class for an object collection is assigned by the management class ACS routine when the first object is stored in the collection.) This can be overridden if you specify a management class when you store the object. Then, the management class ACS routine is run to assign a management class to the object. The management class of an object can be changed with the OSREQ CHANGE macro or through class transition.

You can explicitly specify a management class on the following, but a management class determined by the management class ACS routine takes precedence:

- JCL DD statements
- TSO/E ALLOCATE command
- DFSMSdss COPY and RESTORE commands
- Access method services ALLOCATE, DEFINE, and IMPORT commands
- Dynamic allocation requests through ISPF/PDF data set allocation panels
- OSREQ STORE and OSREQ CHANGE macro requests

End users cannot override any of the attribute values that you assign to a data set or object through the management class ACS routine except for the expiration attributes. Note that this is on the condition that the retention limit is not exceeded (for data sets or objects) or set to NOLIMIT (for data sets). See "Defining Management Class Expiration Attributes" on page 69 for information about the order of precedence for expiration attributes.

The syntax for specifying a management class on a JCL statement is:

```
MGMTCLAS=management-class-name
```

The syntax for specifying a management class on a TSO/E command is:

```
MGMTCLAS(management-class-name)
```

The syntax for specifying a management class on an access method services command is:

```
MGMTCLAS(-) or MANAGEMENTCLASS(management-class-name)
```

The syntax for specifying a management class on an OSREQ STORE or CHANGE macro is:

```
MGMTCLAS=management-class-area or MGMTCLAS=(management-class-area-pointer)
```

For information on determining management classes through ACS routines, see "Chapter 9. Defining ACS Routines" on page 139.

Defining Additional Management Classes

You can copy existing management classes and modify them to create new management classes by using the COPY line operator, which is explained in "Copying SMS Classes, Storage Groups, and Aggregate Groups" on page 204.

Chapter 6. Defining Storage Classes

When attempting to improve performance without SMS, you place important and critical data sets on selected storage devices. If you have data sets that consistently require short response times, you place them on DASD volumes that have low I/O rates or that are connected to cache storage controllers. If you have data sets that require continuous availability, you place duplicate copies of them on other DASD volumes.

SMS uses storage classes to separate data set performance objectives and availability from physical storage. SMS also provides attributes for the following:

- Sequential data striping for batch processing
- Allocation to a volume which supports concurrent copy
- VSAM record-level sharing between different systems with storage classes defined to use the coupling facility (CF).

This chapter describes storage classes and shows you how to define them through ISMF.

Understanding Storage Classes

A storage class is a list of storage objectives and requirements. Each storage class represents a list of services that are available to data sets and objects having similar access requirements. A storage class does not represent any physical storage, but rather provides the criteria that SMS uses in determining an appropriate location to place a data set or object.

In general, SMS attempts to select a location that meets or exceeds the specified objective, but SMS does not guarantee response time. If no location satisfies the performance objective, SMS attempts to find a location that most closely matches the specified objective: on a DASD or tape volume for data sets, or in an object storage hierarchy for objects. For data sets, if there is more than one device of similar characteristics to choose from, SMS selects the volume with the most available space.

Only you, as a storage administrator, can define or alter the storage classes for an SMS complex. End users can specify but cannot override storage classes assigned for a data set by the ACS routines.

Storage Classes for Data Sets

With the exception of tape data sets, a data set is system-managed if it has a storage class. When you assign a storage class to a system-managed DASD data set, SMS places the storage class name in both the BCS and the VVDS catalog entries of the data set.

To satisfy the availability requirements of critical data sets, SMS selects a volume that can provide data access even in the event of a single device failure. For example, this can be satisfied by a device that has dual copy active and that is an array DASD.

To satisfy the availability requirement of standard data sets, SMS selects a volume that does not have the dual copy feature active. Please note that SMS might select a volume that is an array DASD.

Through storage class, you can also specify whether a data set is referenced primarily in the read or write mode.

For information on establishing dual copy volumes and enabling or disabling cache functions, see "Appendix B. SETCACHE Functions and Device Information" on page 295. In this section, you can also find the IBM storage control units that support these features.

Tape data sets can be assigned a storage class, but they are not SMS-managed. Only tape volumes are SMS-managed. Therefore, none of the storage class attributes apply to tape data sets, and tape data sets do not have to be cataloged. If you do catalog a tape data set, the SMS information, such as data class, storage class, and management class is not saved in the catalog. Tape data sets with a storage class can be directed to an SMS-managed tape volume via the storage group ACS routine.

Storage Classes for Objects

For OAM, storage class separates performance objectives from physical storage and determines where the object resides in the storage hierarchy.

OAM uses storage class to logically represent the level of service required by an object. This could result in placement of the object on one of the three types of physical storage in its object storage hierarchy:

- Direct access storage devices (DB2 object tables)
- Optical volumes
- Tape volumes.

The default storage class for an object is defined in the catalog entry for the object collection to which the object belongs. If the default management class is not available from the catalog entry, OAM uses the collection entry in the DB2 table and updates the catalog entry. The default storage class is assigned by the storage class ACS routine when the first object is stored in the collection.

Note that if the catalog entry does not exist, but a DB2 table entry does, this is an error. If neither exist, both are created during a first time store to a new collection, using the storage class, management class, and storage group from the ACS routine.

For a more specific discussion of how to use storage classes with objects, see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support*.

Planning Storage Classes

You should create a storage class for each level of service that is to be provided by the installation. Before you actually define storage classes, you need to identify the available hardware in your installation. Fastest response time and data sharing for VSAM data is obtained from the coupling facility (CF). Fastest response time for DASD data sets is obtained from cached devices (for example, devices behind a

3990 Storage Control with cache active, or behind an IBM RAMAC Virtual Array). Object storage classes should reflect the available hardware in the object storage hierarchy.

Defining Storage Class Attributes

A storage class definition contains both descriptive and access-related information. To identify and refer to storage classes, you assign each one a unique name that contains from one to eight alphanumeric characters. Each storage class definition maintains an owner ID that identifies the storage administrator who originally created or last modified the storage class. Also, each storage class contains an optional 120 character description field for describing its contents.

This section discusses defining storage class attributes. See the following sections for more information:

- Generating lists of storage classes, see “Listing SMS Classes, Aggregate Groups, Storage Groups, and Libraries Using ISMF” on page 190
- Altering a storage class, see “Altering Storage Classes” on page 198
- Displaying storage class information, see *DFSMS/MVS Using ISMF*.

You can use ISMF to define storage class attributes by selecting option 3, Define a Storage Class, from the ISMF Primary Option Menu for Storage Administrators. Figure 32 illustrates the Storage Class Application Selection panel.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                     STORAGE CLASS APPLICATION SELECTION
Command ===>
To perform Storage Class Operations, Specify:
  CDS Name  . . . . . . . 'SMS.SCDS1.SCDS'
                               (1 to 44 character data set name or 'Active')
  Storage Class Name  . . SC01      (For Storage Class List, fully or
                                    partially specified or * for all)
Select one of the following options  :
  3  1. List          - Generate a list of Storage Classes
     2. Display       - Display a Storage Class
     3. Define        - Define a Storage Class
     4. Alter         - Alter a Storage Class
     5. Cache Display - Display Storage Classes/Cache Sets
If List Option is chosen,
   Enter "/" to select option      Respecify View Criteria
                                   Respecify Sort Criteria

If Cache Display is Chosen, Specify Cache Structure Name . .

Use ENTER to Perform Selection;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 32. Defining a Storage Class

To define a storage class:

1. Specify a CDS NAME.

   The CDS NAME must be the name of an SCDS. ISMF primes the field with the last used name. The default is the quoted word 'ACTIVE', which represents the currently active configuration, but you cannot define storage classes to the 'ACTIVE' configuration.

2. Specify the STORAGE CLASS NAME of the storage class that you are defining. ISMF primes the field with the name last used.
3. Select option 3, DEFINE, and press ENTER.

   ISMF displays page 1 of the Storage Class Define panel, shown in Figure 33. On this panel, you select the storage class attributes.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                              STORAGE CLASS DEFINE             Page 1 of 2
Command ===>

SCDS Name . . . . . : USER8.TEST.SCDS
Storage Class Name  : SC1
To DEFINE Storage Class, Specify:
  Description ==> DEFINE STORAGE CLASS
              ==>
  Performance Objectives
   Direct Millisecond Response . . . .            (1 to 999 or blank)
   Direct Bias . . . . . . . . . . . .            (R, W or blank)
   Sequential Millisecond Response . .            (1 to 999 or blank)
   Sequential Bias . . . . . . . . . .            (R, W or blank)
   Initial Access Response Seconds . .            (0 to 9999 or blank)
   Sustained Data Rate (MB/sec)  . . .            (0 to 999 or blank)
  Availability . . . . . . . . . . . . N          (C, P ,S or N)
  Accessibility  . . . . . . . . . . . N          (C, P ,S or N)
   Backup  . . . . . . . . . . . . . .            (Y, N or Blank)
   Versioning  . . . . . . . . . . . .            (Y, N or Blank)

Use ENTER to Perform Verification; Use DOWN Command to View next Page;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 33. Defining Storage Class Attributes, Page 1 of 2

The SCDS Name and Storage Class Name fields are output fields that contain the SCDS and storage class names you specified in the Storage Class Application Selection panel. The Description field is an optional field of 120 characters in which you can describe the storage class.

You can leave the Storage Class Define panel at any time without saving the storage class by issuing the CANCEL command.

The following sections describe the storage class attributes you can define.

Defining Performance Objectives

In the performance objectives fields of the Storage Class Define panel, you can request millisecond response (MSR) times and indicate the bias of both direct and sequential access data sets. All of the performance attributes are optional.

If you leave all MSR and bias fields blank (direct and sequential), SMS ignores device performance during volume selection.

Note that the MSR and bias values you specify in the storage class can be used to determine how buffers are to be allocated when system-managed buffering is used for VSAM applications. To do this, specify a value of `System` for the Record Access Bias attribute in the data class. This capability applies only to system-managed VSAM data sets allocated in extended format and accessed by batch applications.

Defining Millisecond Response Time

The MSR serves two purposes in SMS. First, it is used as the performance objective for selecting candidate volumes for new data set placement. During a new data set allocation, SMS searches for a volume that meets or closely matches this objective. If no volume satisfies the objective, then SMS attempts to find a volume that comes closest to matching it. If more than one MSR is explicitly or implicitly specified, the storage class and associated device MSRs are averaged and compared.

Second, if the data is placed on a volume attached through an IBM 3990 Storage Control with cache, and cache is enabled for that volume, the MSR is used to determine if caching is mandatory, optional, or should be inhibited for the data set. This attribute does not apply to objects.

You can request SMS to ignore various device performances during volume selection by leaving all MSR and BIAS fields blank. This lets you spread data evenly across non-cached and cache active devices.

Millisecond Response Time and Data Set Allocation: DASD can have different performance capabilities for direct access (random access, for example) and for sequential access applications. Its performance capabilities depend on whether you are reading data or writing data.

Each device type and model has a predetermined MSR capability for each condition. Additionally, if the device is attached to a cache capable control unit, the response capabilities are improved when caching is active. Therefore, each device is represented by eight MSR values:

- Uncached Performance
  - Direct Read MSR
  - Direct Write MSR
  - Sequential Read MSR
  - Sequential Write MSR
- Cache performance, DASD Fast Write performance, or both (if active)
  - Direct Read MSR
  - Direct Write MSR
  - Sequential Read MSR
  - Sequential Write MSR

If a device is cache capable, it must also have caching active at the time of allocation in order to be represented by the caching MSR values.

		UNCACHED VALUES[1]				CACHE VALUES				CACHE/DASD FAST WRITE				
		DIRECT		SEQUEN-TIAL		DIRECT		SEQUEN-TIAL		DIRECT		SEQUEN-TIAL		
DEVICE	CONTROLLER	RD.	WT.	RD.	WT.	RD.	WT.	RD.	WT.	RD.	WT.	RD.	WT.	IART[2]
3350 3375	3880-1[4]	30	30	30	30	30	30	30	30	30	30	30	30	0
3380	3880-3[4]	25	25	25	25	25	25	25	25	25	25	25	25	0
	3880-13[3]	25	25	25	25	14	25	14	25	14	25	14	25	0
	3880-23[3]	25	25	25	25	10	25	10	25	10	25	10	25	0
3380 3390-1 3990-2 3390-3	3990-2[4]	25	25	25	25	25	25	25	25	25	25	25	25	0
3380 3390-1 3390-2 3390-3	3990-3/6	25	25	25	25	10	25	10	25	10	6	10	6	0
3390-9	3990-2[4]	50	50	50	50	50	50	50	50	50	50	50	50	0
	3990-3/6	50	50	50	50	35	50	25	50	35	30	25	25	0
9345	9343-x0x[4]	25	25	25	25	25	25	25	25	25	25	25	25	0
	9343-xCx[3]	10	25	10	25	10	25	10	25	10	25	10	25	0
9395	9394	10	6	10	6	10	6	10	6	10	6	10	6	0
9392-1 9392-2 9392-3	3990-3/6	10	25	10	25	10	25	10	25	10	6	10	6	0
9393 9396		10	25	10	25	10	25	10	25	10	6	10	6	0
9397		4	4	4	4	4	4	4	4	4	4	4	4	0
3995-151[4]		600	620	230	250	600	620	230	250	600	620	230	250	10
3995-153[4]		575	590	200	215	575	590	200	215	575	590	200	215	10

Notes:

1. Uncached Values of a device are always used an MSR of 999 is coded.
2 Initial access response time (IART) value = seconds; all other = milliseconds.
3. Does not support DASA FAST WRITE.
4. Does not support CACHE, does not support DASD FAST WRITE.

Figure 34. MSR Capabilities

From the numbers included in Figure 34, you can observe that:

- The fastest write performance that can be currently achieved is 6 milliseconds.
- The fastest read performance that can be currently achieved is 10 milliseconds.
- When cache fast write is not active, write performance is the same as for an uncached device.

You can, for example, help direct allocation to a 3390-9 by specifying an appropriate MSR. A direct MSR of 35 milliseconds with a bias of read would favor a cache-active 3390-9.

Millisecond Response Time and Cache Management: At run time, when control unit caching is available, data is divided into three categories: must cache, may cache, and never cache.

Dynamic data set cache management is an SMS feature that permits expanded use of a storage controller that supports dynamic cache management enhancement (DCME) with cache, caching and DASD fast write features when these features are underused or provides restricted use when they are overused. Only system-managed data sets that reside on volumes attached through a storage controller that supports DCME with cache are affected. Data sets not managed by SMS and data sets that reside on volumes attached through storage controls other than a storage controller that supports DCME with cache are not affected.

Dynamic cache management improves performance when a storage controller that supports DCME with cache is overloaded. It implements a cache management algorithm that optimizes the selection of data sets that are cache candidates. This enhancement also prevents over-commitment of non-volatile storage. New I/O statistics by data set and SMS storage class can be collected and monitored by using the System Management Facility (SMF). Refer to *OS/390 MVS System Management Facilities (SMF)* for details on the structure of SMF type 42 records.

As a function of dynamic data set cache management, each system-managed data set is assigned a cache usage attribute and a DASD fast write usage attribute for sequential and direct accessing modes. These usage attributes are based on the MSR and bias specifications found in the storage classes associated with the data set. These attributes are:

- Cache usage attribute:

  The data set might have one cache usage attribute for sequential accessing mode and a different attribute for direct accessing mode.

  Must Cache
  : The respective MSR specification can be met only through the use of cache. BIAS=Read, Write or blank.

    As the name implies, this is data that must always be cached in order to meet the performance requirement. If you specify an MSR that is lower than any device can provide without caching, then

    – allocation places the data on a device closest to the requested MSR that has cache active (if one is available) and

    – the data is cached at execution time (when cache is active) because the MSR cannot be met by the uncached performance of the device.

    It is possible that as new devices are introduced, equivalent MSR might be obtainable without the use of cache. In that case, the data might be allocated to a non-cached device, but the MSR is still achievable.

  May Cache
  : The respective MSR specification can be met without the use of cache. BIAS=Read, Write or blank.

    Any data that is on a cache capable control unit is considered a candidate for *may cache*. At execution time, if the data is neither must cache nor never cache, dynamic cache management determines if and when to cache the data.

  Never Cache
  : Never Cache data is data that is known to be cache unfriendly. An MSR of 999 indicates that data is never to be cached at execution time, even if it

residesʼ on a cache active device. SMS volume selection prefers a volume whose performance is equivalent to 25ms and has Cache inactive. See Table 3.

- DASD fast write usage attribute:

  The data set might have one DASD fast write usage attribute for sequential accessing mode and a different attribute for direct accessing mode.

  Must DASD Fast Write
  The respective MSR specification cannot be met without the use of DASD fast write. BIAS=Write.

  May DASD Fast Write
  The respective MSR specification can be met without the use of DASD fast write. BIAS=Write, Read or blank.

  Never DASD Fast Write
  The respective MSR specification is 999.

Note: You should direct your IDCAMS query regarding the status of Cache/DFW to the controller, since querying the device can lead to incorrect conclusions.

Defining Bias

Bias determines which volumes MSR performance numbers (read, write, or both) to consider during volume selection. If you specify a read (R) bias, a cache storage control should be available to allow caching. If you specify a write (W) bias, the DASD fast write feature of an IBM 3990 Storage Controller with cache should be available to allow the use of DASD fast write. If you do not specify a value for bias (blank), the MSR time determines whether caching or DASD fast write are used.

Once the data sets have been allocated, the MSR time determines whether caching or DASD fast write are used. You can inhibit caching or DASD fast write by specifying a MSR time of 999. If a bias is specified without an accompanying MSR, an MSR of 6ms for write bias or an MSR of 10ms for read bias is used for volume selection. If the MSR is blank, then the data set is May Cache and May DASD Fast Write.

Note: Allocation does not fail when a storage control device does not have caching or DASD fast write available.

The following chart can help determine which setting ('May, Must, Never') is set by DCME:

Table 3. D/T3990 SMS Cache Candidate Tokens for Sequential and Direct Requests

		Direct Cache Token		Sequential Cache Token	
Direct/Sequential MSR	**Dir/Seq Bias**	**Read**	**Write[3]**	**Read**	**Write[3]**
999	any	Never	Never	Never	Never
blank	blank	May	May[2]	May[1]	May
MSR < UCr[4] & MSR < UCw[5]	blank	Must	May[2]	Must	May
MSR < UCr & MSR >= UCw	blank	Must	May[2]	Must	May
MSR >= UCr & MSR < UCw	blank	May	May[2]	May[1]	May
MSR >= UCr & MSR >= UCw	blank	May	May[2]	May[1]	May
blank	Read	May	May[2]	May[1]	Never

Table 3. D/T3990 SMS Cache Candidate Tokens for Sequential and Direct Requests (continued)

		Direct Cache Token		Sequential Cache Token	
Direct/Sequential MSR	**Dir/Seq Bias**	**Read**	**Write[3]**	**Read**	**Write[3]**
MSR < UCr & MSR < UCw	Read	Must	May[2]	Must	Never
MSR < UCr & MSR >= UCw	Read	Must	May[2]	Must	Never
MSR >= UCr & MSR < UCw	Read	May	May[2]	May[1]	Never
MSR >= UCr &MSR >= UCw	Read	May	May[2]	May[1]	Never
blank	Write	May	May[2]	May[1]	May
MSR < UCr[4] &MSR < UCw[5]	Write	Must	Must	Must	Must
MSR < UCr & MSR >= UCw	Write	Must	May[2]	Must	May
MSR >= UCr &MSR < UCw	Write	May	Must	May[1]	Must
MSR >= UCr & MSR >= UCw	Write	May	May[2]	May[1]	May

Notes:

1. The 3990 Extended Platform results in any sequential read request to always be cached unless MSR 999 is specified.
2. The 3990 Extended Platform results in all direct write requests from VSAM or Media Manager to be treated as must write.
3. All write requests to a RAMAC device are treated as must write.
4. Uncached values read performance.
5. Uncached values write performance.

Refer to Figure 34 on page 88 for performance characteristics of a particular IBM device.

Set the storage class to indicate the desired performance for a data set. SMS volume selection and DCME work together to provide the closest performance possible at the time of allocation and open.

Defining Initial Access Response Seconds

You can use the Initial Access Response Seconds attribute to optionally specify the desired response time (in seconds) that is required to locate, mount and prepare a piece of media for data transfer. OAM uses this value to interpret storage level, that is, to place an object at an appropriate level in the object storage hierarchy. For objects, both the initial access response time (IART) and sustained data rate (SDR) are applicable.

OAM uses IART as follows: if the value is 0, OAM writes to DASD; if the value is 1–9999, OAM selects removable media (either optical or tape). When combining IART with SDR, OAM writes to tape when the SDR value is less than or equal to 3; when the SDR value is greater than 3, OAM writes to optical.

The IART for an optical volume depends on its location. The time required for an operator to mount a shelf-resident optical volume is significantly longer than the time for the automatic mounting of a library-resident optical volume in an optical library. You can use from one to four characters to specify a valid value of 0 to 9999 or you can leave the field blank. The default is blank and selects fixed DASD. However, any OAM request that tries to use a storage class with a blank IART value fails.

To allow mountable DASD devices to be selected for new allocations of non-objects, you must specify an IART value from 1 to 9999. Specifying 0 or blank rejects mountable DASD devices from the volume selection candidate lists. Since objects are written to DB2 tables which are preallocated, new allocations are done by DB2.

Fixed DASD has a zero IART because the volume is permanently mounted. However, certain optical devices (like 3995-153 or 3995-151) appear to application programs just like DASD devices, but they do have a real IART. Therefore, if the storage class specifies a non-zero IART, though mountable DASD is preferred, fixed DASD is used if it is not possible to select a mountable device. Because eligible storage groups can contain mountable and fixed DASD, and fixed devices can be used for non-zero IART requests, you might want to review your MSR settings.

Defining Sustained Data Rate

You can use the Sustained Data Rate (SDR) attribute to request sequential data striping for batch processing. Striping allows you to spread data across different DASD controllers, and should be completely transparent to the application. The number of stripes is the number of volumes on which the data set is allocated. Striped data sets must be SMS managed. All stripes for a particular data set must reside in the same SMS storage group.

OAM uses SDR to determine whether writes to removable media are for optical or tape media. The IART must be greater than 0.

The system uses the SUSTAINED DATA RATE attribute in conjunction with the device transfer rate to derive the number of stripes a striped data set can have (maximum of 16). The SUSTAINED DATA RATE field specifies the target throughput rate. Valid values for this attribute are blank and 0 to 999.

For non-guaranteed space requests, if the SUSTAINED DATA RATE field is blank or 0, the target number of stripes is 1. If the SUSTAINED DATA RATE value is greater than 1, it is divided by 4 for 3390s, or by 3 for 3380s to determine the stripe count. For example, for a sustained data rate of 24, a storage group of 3380s would have a stripe count of 8, a storage group of 3390s would have a stripe count of 6. The volume must be able to satisfy the primary space requested divided by the number of stripes and not exceed the high threshold specified in the storage group.

Depending on the number of eligible volumes in the selected candidate storage groups, the actual number of stripes allocated to a data set might be less than the derived target number of stripes. For example, if you request a stripe count of 5 and only 3 volumes are available behind two unique controllers, the stripe count is reduced to 3. As the stripe count is decreased, the primary space amount for each stripe increases, except for guaranteed space requests with volume serial numbers.

The Sustained Data Rate attribute is ignored if a data set name type of extended required or extended preferred is not specified in the data class. The Sustained Data Rate attribute is also ignored if you specify Guaranteed Space = YES, in which case the number of volume serial numbers or VOLCNT specified is the number of stripes.

When no volumes supporting striping are available, the data set is allocated non-striped if EXT=P is specified in the data class; the allocation fails if EXT=R is specified in the data class.

See *DFSMS/MVS Using Data Sets* for more detailed information on using SMS to assign storage classes that specify striped data sets.

See “Striping Volume Selection” on page 110 for more information.

Defining Availability

The Availability field is used to specify whether data set access should continue in the event of a single device failure. Storage classes with a blank Availability field default to NOPREF.

CONTINUOUS
Specify an availability of CONTINUOUS if you do not want a device failure to affect processing. Only duplexed and RAID volumes are eligible for this setting.

If CONTINUOUS availability is specified, data is placed on a device that can guarantee that it can still access the data in the event of a single device failure. This option can be met by

- A dual copy volume
- An array DASD

PREFERRED
Array DASD volumes are preferred over non-duplexed volumes. Dual Copy volumes are not candidates for selection.

STANDARD
If data sets do not require such a high level of availability, specify STANDARD availability, which represents normal storage needs.

Specify an availability of STANDARD to cause processing of a data set to stop after a device failure. Simplex volumes are preferred over array DASD. SMS selects only volumes that are *not* dual copy. This attribute does not apply to objects. Array DASD are acceptable candidates for both STANDARD and CONTINUOUS availability requests.

Prior to DFSMS/MVS 1.2, the old default of STANDARD acts the same as the default of NOPREF. Change STANDARD to NOPREF for compatibility with DFSMS/MVS releases prior to DFSMS/MVS 1.2.

NOPREF
Simplex and array DASD are equally considered for volume selection. NOPREF is the default. Dual copy volumes are not candidates for selection.

Note: On DFSMS/MVS 1.2 systems, SMS interprets a blank Availability field as STANDARD instead of NOPREF. To ensure that storage classes are set correctly on a DFSMS/MVS 1.2 system, issue ISMF ALTER and SAVE commands against those storage classes which have a blank Availability field or which specify NOPREF. You do not need to modify the storage classes.

| ## Defining Accessibility

| The storage class accessibility attribute defines the function of the hardware
| supporting point-in-time copy, using either concurrent copy or virtual concurrent
| copy. Point-in-time copy allows database management systems (DBMSs) to take
| what appears to be an instantaneous copy of data, or a ″fast″ point-in-time copy.
| The copy can be for backup purposes (generally to tape) or for copying a database

from one set of DASD volumes to another. The accessibility attribute allows you to direct allocation of new data sets to DASD connected to an IBM 3990 Storage Control unit with cache that supports the concurrent copy function, as well as to IBM RAMAC Virtual Array devices with SnapShot copy support and DFSMSdss with virtual concurrent copy SPE installed.

When you specify accessibility attributes, you identify whether you want SMS to use *versioning* or *backup* devices for either concurrent copy or virtual concurrent copy.

Versioning Device
: can create a ″fast″ point-in-time version of a data set, which is then available for application testing, reporting, or backup operations. While the version is being made, the data set is unavailable for normal application processing for a minimal period of time. Versioning is done using the SnapShot feature of the RAMAC Virtual Array.

Backup Device
: can create a ″fast″ point-in-time backup copy of a data set. While the backup copy is being made, the data set is unavailable for normal application processing for a minimal period of time. Two methods are supported:

  Method 1
  : Establish a concurrent copy session with the 3990 DASD controller and make the backup copy.

  Method 2
  : Use virtual concurrent copy. DFSMSdss uses the SnapShot feature of the RAMAC Virtual Array to create a point-in-time copy and then does I/O to create the backup to whatever target device you specified.

You use a combination of values for the accessibility attribute and its subparameters, Versioning and Backup, to request point-in-time copy. In the Accessibility field, you specify whether allocation to a point-in-time copy-capable volume is required (CONTINUOUS), preferred (CONTINUOUS PREFERRED), or discouraged (STANDARD). You then specify values for the Versioning and Backup subparameters to select which devices you want used for the copy.

The following defines your allocation request to a point-in-time copy-capable volume:

CONTINUOUS (C)
: Only point-in-time copy volumes are selected.

CONTINUOUS PREFERRED (P)
: Point-in-time copy volumes are preferred over non-point-in-time copy volumes.

STANDARD (S)
: Non-point-in-time copy volumes are preferred over point-in-time copy volumes.

NOPREF (N)
: Point-in-time copy capability is ignored during volume selection. This is the default.

The following table identifies the values you specify based on your specific accessibility requirements:

Table 4. Combinations for Requesting Point-in-Time Copy Devices

If these are your choices for accessibility devices:			Specify the following values:
First Choice	**Second Choice**	**Third Choice**	

Versioning device	None	None	`Accessibility = C`
			`Versioning = Y`
			`Backup = N`

Method 1 backup device	None	None	`Accessibility = C`
			`Versioning = N`
			`Backup = Y`

Any versioning or backup device	None	None	`Accessibility = C`
			`Versioning = blank`
			`Backup = blank`

Any versioning or backup device	Any non-accessibility device	None	`Accessibility = P`
			`Versioning = blank`
			`Backup = blank`

Versioning or Method 2 backup device	Method 1 backup device	Any non-accessibility device	`Accessibility = P`
			`Versioning = Y`
			`Backup = Y`

Versioning device	Any non-versioning device	None	`Accessibility = P`
			`Versioning = Y`
			`Backup = N`

Any non-accessibility device	Any versioning or backup device	None	`Accessibility = S`
			`Versioning = blank`
			`Backup = blank`

Any device	None	None	`Accessibility = N`
			`Versioning = blank`
			`Backup = blank`

You can also use the data set alter application to alter a storage class so that data sets can be allocated on point-in-time copy-capable volumes.

Defining Guaranteed Space

You can allocate space for single volume and multivolume data sets before the job step runs by specifying a storage class with a Guaranteed Space attribute. SMS fails the request if sufficient space is not available. You can also use the Guaranteed Space attribute to allocate a data set on specific volumes. The Guaranteed Space attribute does not apply to objects, but it does apply to both VSAM and non-VSAM managed data sets.

For a multivolume system-managed data set, primary space is preallocated on all the volumes. The first volume becomes the primary volume. All remaining volumes become candidate volumes with preallocated space. When the primary extent on the current volume becomes full, VSAM attempts to create secondary extents on the current volume. If not enough space is left on the current volume for it to be extended, VSAM then uses the preallocated primary extent on the next volume. This next volume is converted from a candidate volume with space to a primary volume.

Generally, for VSAM key-sequenced data sets with key ranges specified, space is preallocated on as many volumes as key ranges specified. For more information on key ranges, see the description of the DEFINE CLUSTER command in *DFSMS/MVS Access Method Services for ICF*.

Preallocating Space for Multivolume Data Sets

The following JCL allocates a multivolume data set on volumes selected by SMS. Storage class DBLOG must have Guaranteed Space=Y. This example allocates 100MB on each of five volumes. When all of the allocated space is used for the data set on one volume, the secondary space is allocated as required in extents on that volume.

```
//DD1  DD  DSN=ENG.MULTFILE,DISP=(,KEEP),STORCLAS=DBLOG,
//         SPACE=(1,(100,25)),AVGREC=M,
//         UNIT=(3380,5)
```

Figure 35. Example of JCL to Allocate a Multivolume Data Set On SMS-Selected Volumes

1. After 100MB is used on the first volume, 25MB extents of secondary space is allocated on it until the extent limit is reached or the volume is full.
2. If more space is needed, 100MB of primary space is used on the second volume. Then, more secondary space is allocated on that volume.
3. The same process is repeated on each volume, as shown in Figure 36.

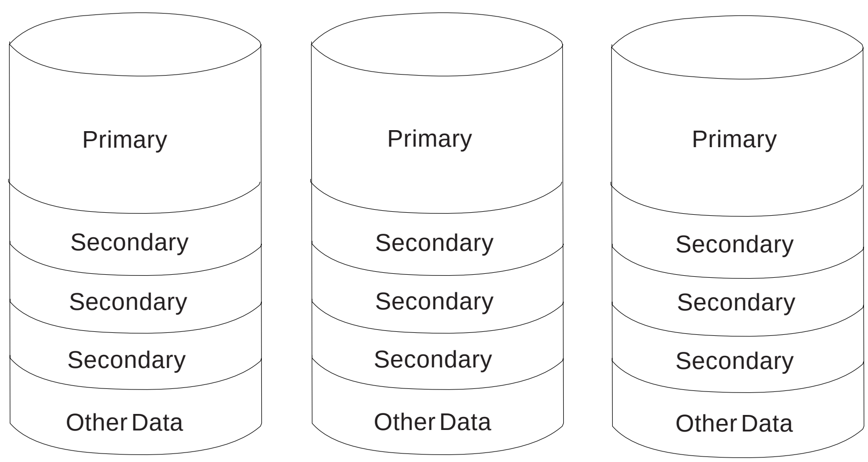

Figure 36. Allocation of Primary and Secondary Space for Multivolume Data Sets

Honoring Specific Volume Requests

In addition to preallocating space for multivolume data sets, you can use the Guaranteed Space attribute to let SMS honor a request for single or multiple volumes explicitly specified by the user. For example, you can allow users to preallocate a non-VSAM multivolume data set on specific volumes. SMS insures that all the specified volumes are in the same storage group and also that the storage group containing the volumes is one of the storage groups assigned to the storage group ACS routine. You can also use the Guaranteed Space attribute to allocate space specifically on volumes within the same storage group. Figure 37 on page 97 shows JCL preallocating space for a multivolume data set on volumes VOL001, VOL002, VOL003, VOL004, and VOL005 if the storage class SPECIAL has Guaranteed Space:

```
//DD1  DD  DSN=ABCD.MULTFILE,DISP=(,KEEP),STORCLAS=SPECIAL,
//         SPACE=(1,(100,75)),AVGREC=M,
//         VOL=SER=(VOL001,VOL002,VOL003,VOL004,VOL005)
```

Figure 37. Example of JCL Preallocating Space for a Multivolume Data Set

This JCL allocates 100MB on all specified volumes for the data set. If the data set disposition is NEW or MOD, when all the primary space is used on the first volume, VOL001, secondary extents of 75MB are obtained up to 15 times on the first volume before starting to fill primary space on the second volume, VOL002. This pattern is repeated for every volume of the data set. If the data set disposition is not NEW or MOD, primary space on all volumes is filled before secondary extents of 75MB are obtained up to 15 times, starting on the last volume, VOL005.

If you specify NO for Guaranteed Space, then SMS chooses the volumes for allocation, ignoring any VOL=SER statements specified on JCL. Primary space on the first volume is preallocated. NO is the default.

End users can allocate space on specific volumes for a data set if you:

- Create at least one storage class with the Guaranteed Space attribute
- Ensure the user is authorized to the storage class with the Guaranteed Space attribute
- Write a storage group ACS routine that assigns a storage group that contains the volumes explicitly specified by the user
- Ensure all volumes explicitly specified by the user belong to the same storage group, by directing an allocation which is assigned a Guaranteed Space storage class to all the storage groups in the installation.
- Ensure the requested space is available since there is no capability in SMS to allow specific volume requests except with the Guaranteed Space attribute.
- Ensure availability and accessibility specifications in the storage class can be met by the specified volumes.

Defining Guaranteed Synchronous Write

You can use the Guaranteed Synchronous Write attribute to indicate whether your system should return from a BSAM CHECK (or WAIT) issued for a WRITE against a PDSE member before (unsynchronized) or after (synchronized) the data has actually been written to DASD. This attribute does not apply to objects.

Specify a Y for synchronized write or an N for no synchronization.

Defining Use of the Coupling Facility for VSAM Record-Level Sharing

This section describes how to associate a storage class with a CF cache set defined in the base configuration, thereby making any data set associated with the storage class eligible for VSAM record-level sharing. It also describes how to assign a weight value to the data, so as to indicate its relative importance.

1. Press the DOWN key or use the DOWN command to view the second page of the Storage Class Define panel, shown in Figure 38 on page 98:

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                            STORAGE CLASS DEFINE              Page 2 of 2
Command ===>

SCDS Name . . . . . : USER8.TEST.SCDS
Storage Class Name  : SC1

To DEFINE Storage Class, Specify:

  Guaranteed Space . . . . . . . . . N           (Y or N)
  Guaranteed Synchronous Write . . . N           (Y or N)
  CF Cache Set Name  . . . . . . . .             (up to 8 chars or blank)
  CF Direct Weight . . . . . . . . .             (1 to 11 or blank)
  CF Sequential Weight . . . . . . .             (1 to 11 or blank)






Use ENTER to Perform Verification; Use UP Command to View previous Page;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 38. Defining Storage Class Attributes - Page 2

2. Enter the name of a CF cache set defined in the base configuration.

   When you specify a cache set name, any data set associated with this storage class becomes eligible for record-level sharing using the CF. CACHE SET NAME maps the storage class to a CF cache set defined in the SMS base configuration, for which CF cache structures have been defined.

   In a JES3 environment, be careful to define cache set names only in those SMS storage classes that are used by data sets opened for VSAM RLS processing. When you define a cache set name in a storage class, any job accessing a data set associated with that storage class is scheduled on a VSAM RLS-capable system. If all storage classes have cache set names defined for them, then all jobs accessing SMS-managed data sets are scheduled to VSAM RLS-capable systems. This could cause a workload imbalance between those systems and down-level systems.
3. Specify a weight attribute indicating the data's relative importance in the CF DIRECT WEIGHT or the CF SEQUENTIAL WEIGHT fields. Use the CF DIRECT WEIGHT field for direct data; use the CF SEQUENTIAL WEIGHT field for sequential data. The default is a weight value of 6.

   Note: DFSMS/MVS only supports the default value. All data is assigned a weight value of 6 regardless of the value you specify.

   See "Chapter 14. Administering VSAM Record-Level Sharing" on page 229 for more information about administering VSAM record-level sharing.

Assigning Storage Classes

With the exception of tape data sets, a data set is SMS-managed if it is assigned a storage class. You can assign storage classes either through the storage class ACS routine or by explicit specification. If you do not specify an explicit storage class when you store an object, the object is assigned the storage class that is defined in the collection to which the object belongs. The default storage class for an object collection is assigned by the ACS routine when the first object is stored in that collection. If the storage class ACS routine determines a storage class, it takes precedence over one that is explicitly specified by any of the following:

- JCL DD statements
- TSO/E ALLOCATE command
- DFSMSdss COPY and RESTORE commands
- Access method services ALLOCATE, DEFINE, and IMPORT commands
- Dynamic allocation requests, such as with ISPF/PDF data set allocation panels
- OSREQ STORE and CHANGE requests

ACS routines are required for Distributed FileManager/MVS-created data sets to ensure that they are SMS-managed. These data sets must be SMS-managed. If a remote application attempts to create a local data set in non-SMS-managed storage, Distributed FileManager/MVS refuses to honor the request because it only creates SMS-managed data sets. Distributed FileManager/MVS does, however, support the access of non-SMS-managed data sets.

The syntax for specifying a storage class on a JCL statement is

```
STORCLAS=storage-class-name
```

The syntax for specifying a storage class on a TSO/E command is:

```
STORCLAS(storage-class-name)
```

The syntax for specifying a storage class on an access method services command is:

```
STORCLAS(-)
```

The syntax for specifying a storage class on an OSREQ STORE or CHANGE macro is:

```
STORCLAS=storage-class-area or STORCLAS=(storage-class-area-pointer)
```

For information on determining storage classes with ACS routines, see “Chapter 9. Defining ACS Routines” on page 139.

Defining Additional Storage Classes

You can copy existing storage classes and modify them to create new storage classes by using the COPY line operator, which is explained in “Copying SMS Classes, Storage Groups, and Aggregate Groups” on page 204.

Understanding Volume Selection

SMS classifies all volumes in the selected storage groups into four volume categories:

Primary

Volumes that meet all the specified criteria in the storage class in addition to the volumes being online and below threshold. Both the volume status and storage group status are enabled.

Volume selection starts from this list.

Secondary
Volumes that do not meet all the criteria for primary volumes. If there are no primary volumes, SMS selects from the secondary volumes.

Tertiary
Volumes are marked tertiary if the number of volumes in the storage group is less than the number of volumes requested. If there are no secondary volumes available, SMS selects from the tertiary candidates.

Rejected
Volumes which do not meet the *required* specifications (ACCESSIBILITY = CONTINUOUS, AVAILABILITY = STANDARD or CONTINUOUS, ENABLED or QUIESCED, ONLINE...). These volumes are marked rejected and are not candidates for selection.

This section describes how SMS chooses DASD volumes. It also describes the two types of volume selection: conventional volume selection and striping volume selection. Finally, it discusses why volume selection might fail.

How SMS Selects Volumes for Data Set Allocation

Figure 39 on page 101 shows how SMS selects volumes to allocate data sets that are not striped.

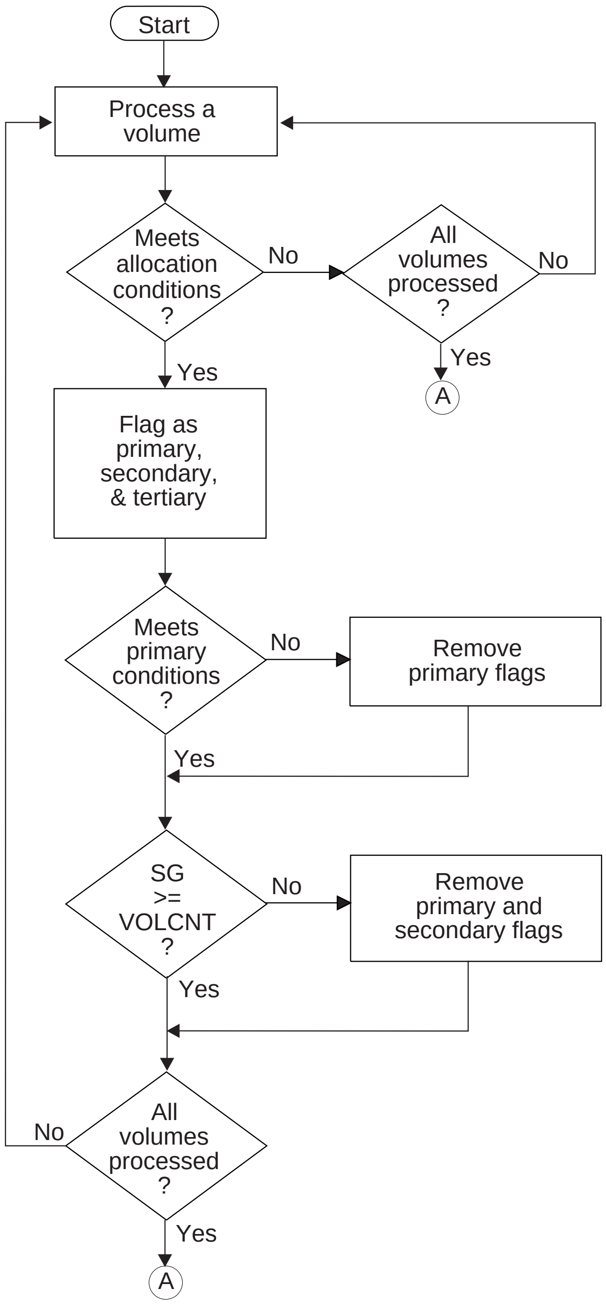

Figure 39. How SMS Selects Volumes for Data Set Allocation (Part 1)

When the processing shown in Figure 39 completes, SMS proceeds as shown in Figure 40 on page 102:

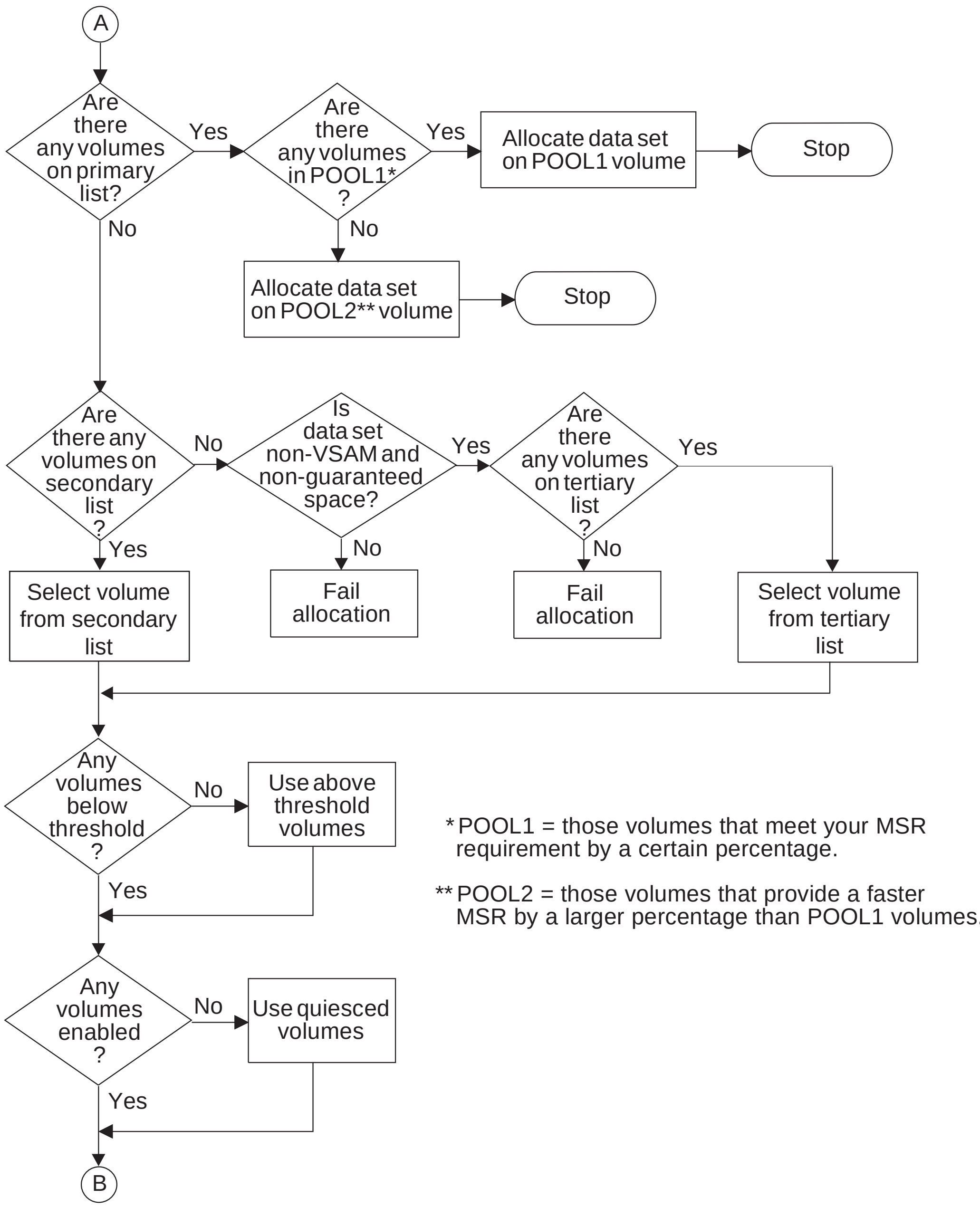

Figure 40. How SMS Selects Volumes for Data Set Allocation (Part 2)

When the processing shown in Figure 40 completes, SMS proceeds as shown in Figure 41 on page 103:

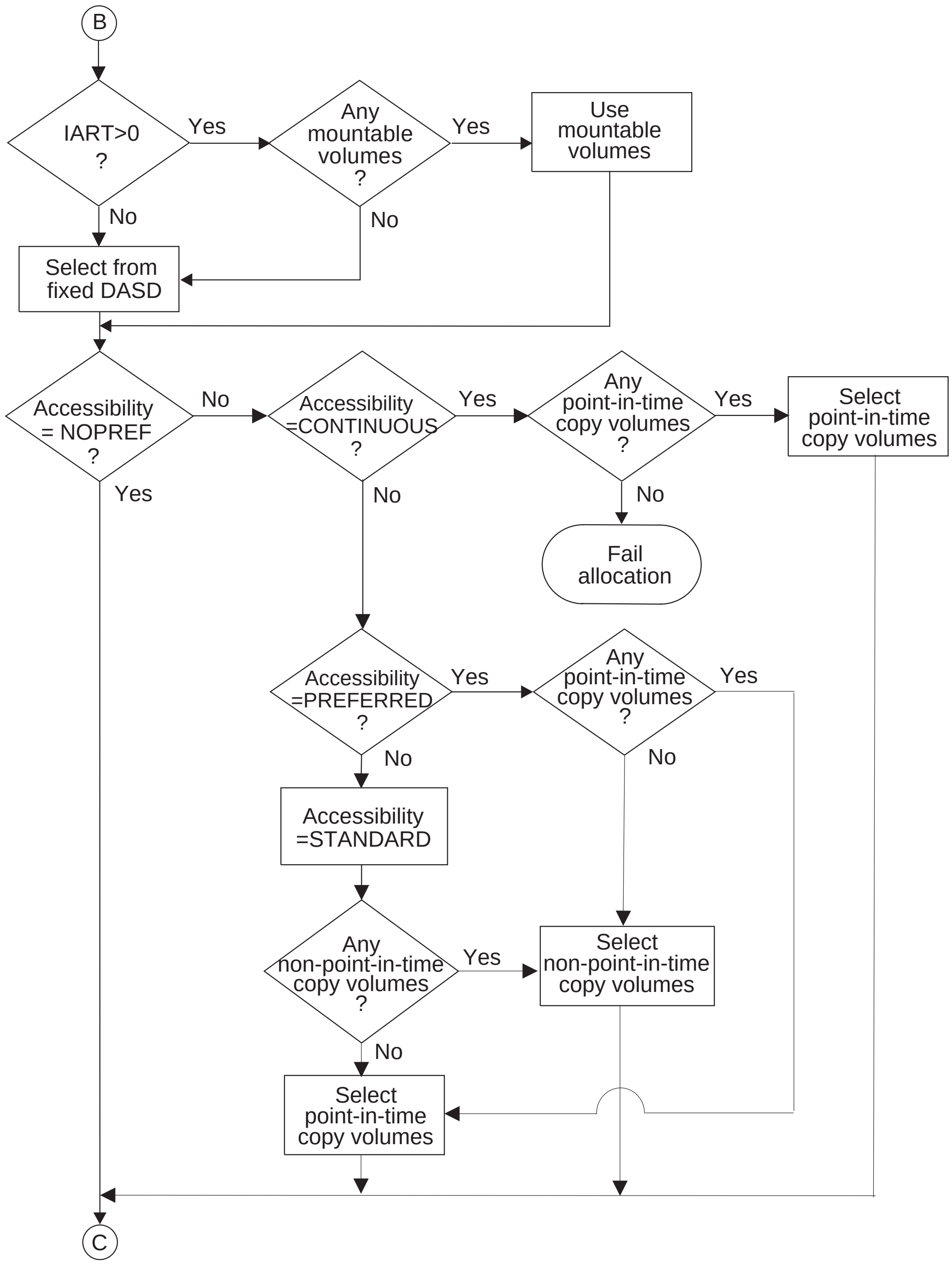

Figure 41. How SMS Selects Volumes for Data Set Allocation (Part 3)

When the processing shown in Figure 41 completes, SMS proceeds as shown in Figure 42 on page 104:

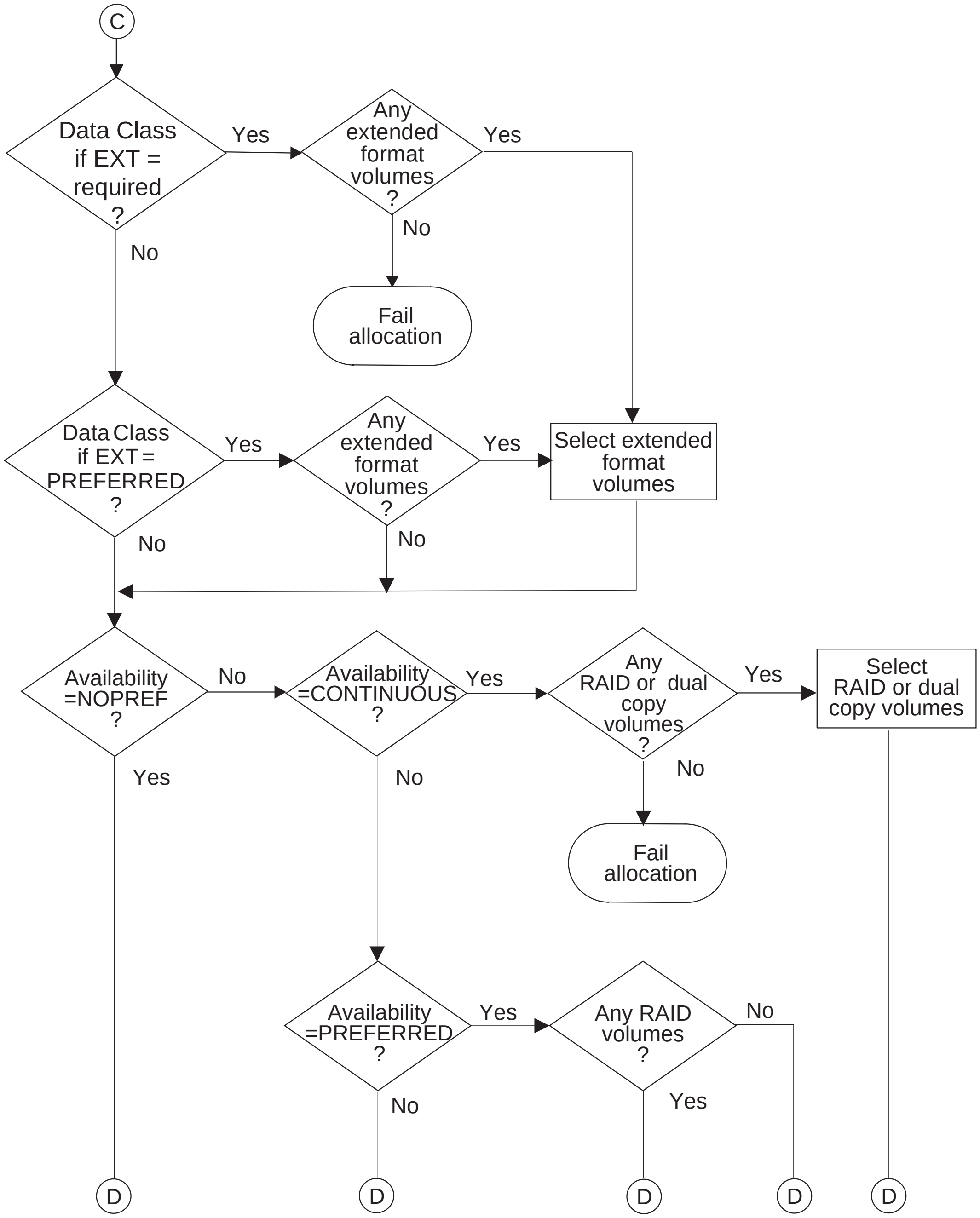

Figure 42. How SMS Selects Volumes for Data Set Allocation (Part 4)

When the processing shown in Figure 42 completes, SMS proceeds as shown in Figure 43 on page 105:

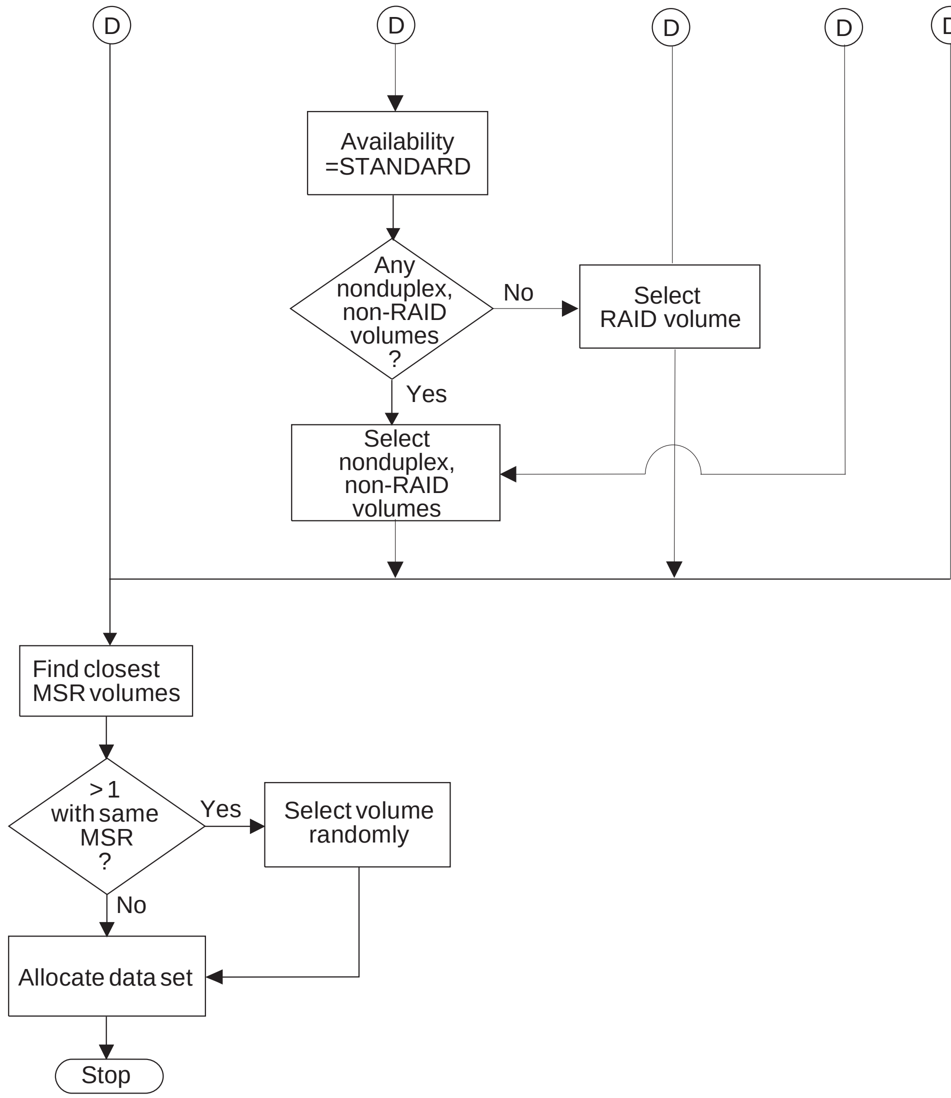

Figure 43. How SMS Selects Volumes for Data Set Allocation (Part 5)

After the system selects the primary space allocation volume, that volume's associated storage group is used to select any remaining volumes requested for the data set.

You can add new devices into an existing MVS complex and take advantage of different performance and device characteristics.

Conventional Volume Selection

Conventional volume selection is used for all non-striped data sets, as well as for data sets allocated with an SDR or zero or blank. The following sections describe the lists used for data placement: primary, secondary and tertiary.

The Primary List

All the volumes in all the specified storage groups are candidates for the first, or primary list. The primary list consists of those devices that fit all the storage class requirements. All devices on this list are considered equally qualified to satisfy the request.

Whenever possible, a choice is made from among the candidates on this list, essentially choosing the device that is the least busy. It is possible that there are no volumes on the primary list if no devices in the storage group(s) meet all of the primary list criteria. It is also possible that even though there are devices on this primary list, the data set cannot be successfully allocated to any of these devices (i.e., there is not enough space available when the actual allocation request is made.)

For example, if you specify a 25 MSR in your storage class, volumes that are close to an MSR of 25 would be POOL1 volumes. POOL1 volumes are those volumes that meet your MSR requirement by a certain percentage. Each MSR percentage represents a range of MSRs. All volumes that fall within an MSR percentage are considered equal in performance. Volumes that provide a faster MSR by a larger percentage than POOL1 volumes are considered POOL2 volumes. POOL2 volumes include POOL1 volumes plus the volumes in the next higher MSR percentage. POOL1 volumes are preferred over POOL2 volumes.

Primary volumes include the POOL1 and POOL2 volumes that can meet the following criteria:

- SMS storage group and volume status of enabled
- MVS status of online
- IART requirement
- The number of volumes in the storage group satisfies the volume count
- Accessibility requested
- Availability requested
- Guaranteed space requirement
- The volumes can allocate the request without exceeding high threshold
- The volume supports extended format if EXT = PREFERRED or REQUIRED is requested in the data class.

The primary volume list is passed to the system resource manager (SRM) which selects a volume based on channel resources.

The Secondary List

The secondary list contains all the volumes that can be used for data set allocation, even if they don’t meet all the storage class criteria. It also includes all primary volumes.

If allocation is not successful from the primary list, then it is attempted from this list, starting at the top and working down to the bottom until a successful allocation is achieved. The order of this secondary list has been established to address the following in preferred order:

1. ABEND X37 prevention

   Volumes that are below threshold are placed ahead of volumes that are above threshold.

2. Spill/Overflow

   Enabled volumes are placed ahead of quiesced volumes. (For information regarding spill volumes and their role in volume selection, see *MVS/ESA SML: Managing Storage Groups*.)

3. Volume characteristics

   Volume characteristics include availability, accessibility, mount times, guaranteed space, and extended format.

4. Performance

   Volumes are ordered working outward from the requested MSR, first to the faster performance volumes, followed by the slower performance volumes.

The Tertiary List

When a storage group does not contain enough volumes to satisfy the volume count, all volumes in the storage group are flagged as tertiary. Tertiary volumes are only selected when there are no primary or secondary volumes and the request is for a non-VSAM non-GUARANTEED SPACE request. The concept of tertiary volumes does not apply to VSAM data sets. In other words, for all VSAM non-GUARANTEED SPACE requests, the volume count does not play a role in determining which storage group is selected.

Why a Volume is Not Placed on the Primary List

A volume is not placed on the primary list for one of the following reasons:

- The volume was rejected. See “Some Possible Reasons for Failure in Volume Selection” on page 112 for more information.
- The volume could not allocate the primary space request without going over the high threshold specified in the assigned storage group.
- The volume or storage group is not enabled (per SMS status).
- The volume did not satisfy the requested MSR performance.
- The VTOC index is broken or disabled, resulting in SMS not being updated with space statistics from the Common VTOC Access Facility (CVAF). This can also happen if OEM products bypass CVAF, the component that notifies SMS of space changes. SMS not being updated with space statistics can also result in a volume being overused.
- A non-zero IART value was specified in the storage class to remove fixed DASD from the primary list.
- An accessibility value of PREFERRED was specified in the storage class, barring all volumes that do not have 3990 extended platform support from the primary list.
- An accessibility value of STANDARD was specified in the storage class, barring all volumes that have 3990 extended platform support from the primary list.
- The data class specifies ’extended format preferred’ and the volume does not support ’extended format.’ In this case, the volume is not placed on the primary list but is NOT rejected. If the data class specifies ’extended format required’ any volume that does not support ’extended format’ is rejected.
- The volume was placed on the tertiary list because the number of volumes in the storage group was less than the number of volumes requested. (See “The Tertiary List” for more information on the tertiary list.)

  An example of this is when a tape request of 5 volumes causes a tape mount management buffer storage group containing 2 volumes to be marked tertiary while a spill (quiesced) storage group containing 5 volumes is to be marked secondary. This results in the quiesced volumes being preferred before the tape mount management volumes.

- The controller was IMLed while online to MVS. This can result in the MVS device control blocks not reflecting the current state of the volume. The device or devices should be varied offline and back online to update the MVS control block status.
- RAID devices are eliminated from the primary list when STANDARD is specified. Use NOPREF, the AVAILABILITY default, to allow both SIMPLEX and RAID devices to be placed on the primary list.

Why Some Secondary or Tertiary List Volumes Are Preferred

This section describes why some secondary or tertiary volumes are preferred over others. You can use this section to help you determine why there are no volumes on the primary list or why the volume you believe should have been selected was not on the primary list.

Volumes might be preferred for the following reasons, and in the following order of preference:

1. High threshold (space)

   A volume below high threshold is preferred over a volume that is above high threshold. A volume is considered below high threshold when the primary space allocation does not cause the high threshold limit to be exceeded.

2. SMS status

   An enabled volume is preferred over a quiesced volume (if a storage group is quiesced, all enabled volumes in the storage group are also considered to be quiesced). This allows for implementation of spill/overflow volumes/storage groups.

```
--------------------------
                 | Enabled
Below Threshold  |---------
                 | Quiesced
--------------------------
                 | Enabled
Above Threshold  |---------
                 | Quiesced
--------------------------
```

   For more information on spill/overflow volumes and storage groups, see *MVS/ESA SML: Managing Storage Groups*.

3. IART greater than zero

   If you specify an IART greater than zero, mountable volumes (optical) are preferred over fixed DASD. The threshold and status categories are again divided, as shown below:

```
--------------------------------------
                 |          | Mountable
                 | Enabled  |-----------
                 |          | Fixed
Below Threshold  |---------------------
                 |          | Mountable
                 | Quiesced |-----------
                 |          | Fixed
--------------------------------------
  Continued..    |          | Mountable
```

4. Accessibility value

If you specify ACCESSIBILITY=PREFERRED, those volumes that support concurrent copy are preferred over volumes that do not support concurrent copy.

Note: All volumes behind a 3990-6 or a 3990-3 controller (with extended platform microcode) are considered to support concurrent copy.

The NOPREF value can be entered for accessibility and allows both extended platform and non-extended platform volumes to be treated equally (included in the primary list). For a multivolume data set, if any extent resides on a non-extended platform, the entire data set is not accessible for concurrent copy.
|

5. If EXT data class value

   If you specify ’If EXT=P’ in the data class, those volumes that support Extended Format are preferred over volumes that do not support Extended Format.

6. Availability value

   When you specify AVAILABILITY=PREFERRED, Array DASD volumes are placed ahead of non-duplexed volumes (within the next division hierarchy above).

7. MSR value

   The volume that best meets the requested MSR in the storage class is preferred.

8. Never-Cache

   If you specify an MSR of 999 in the storage class, all non-cache active volumes are placed ahead of cache active volumes.

Tuning Considerations

If a storage class is assigned to data sets that occupy 30% of the total space of all selected storage groups, then the selected storage groups (when combined) should have 30% of their space backed by volumes that match all the storage class parameters (such as MSR, accessibility, IART).

Note: Be careful when tuning your storage class ACS routines to match the mixture of device capabilities assigned by your storage group ACS routines. For example, if you have both cached and non-cached 3390-3 in the assigned storage groups and your ACS routine assigned the same storage class (MSR=25) to all data sets, the non-cached devices are preferred over the cached devices. This results in the non-cached devices being overused and the cache-active devices are not selected until all the non-cached devices are above high threshold. If an MSR=2 is specified, then the cached devices are overused.

This also holds true for other attributes. If you always assign ACCESSIBILITY=PREFERRED, then concurrent copy volumes are overused.

Take care when specifying multiple preferences (AVAILABILITY=P, ACCESSIBILITY=P, MSR/BIAS value specified, If EXT=P, IART>0). Each additional preference specified adds additional work to SMS volume selection routines and might affect allocation performance.

Spreading Allocations Across Multiple Volumes.

You might want to spread allocations across multiple volumes to reduce I/O contention and improve performance. After analyzing your data, you might find that certain data may not need some of the advanced storage features in your environment.

If you want to spread conventional allocations across volumes of different features, use one or all of the following suggested parameter settings in the assigned storage and data classes:

Storage Class Parameters
- MSR = blank
- BIAS = blank
- ACCESSIBILITY = NOPREF
- AVAILABILITY = NOPREF
- GUARANTEED SPACE = N

Data Class Parameter
- if EXT = blank

Striping Volume Selection

Striping volume selection is entered only in the following situations:

- During initial allocation:

  Only for non-VSAM data sets where the data class specifies extended format as either ’required’ or ’preferred’ and the SDR in the storage class is non-zero. If the SDR is zero, a non-striped, extended format data set is allocated and conventional volume selection is used. Note that a non-striped extended format data set might also be referred to as a single-striped data set.
- During restore/recall processing:

  If the data set was a multistripe data set when it was migrated or backed-up. If the data set was single-stripe when migrated/backed-up, it follows the conventional volume selection path.

Conventional volume selection is used in the following instances:

- For VSAM striped data sets, which are always single-striped and follow all the rules of non-striped data sets
- When recalling and recovering single-striped multivolume SAM data sets

Volumes are classified as primary and secondary, and primary volumes are preferred over secondary volumes. A single primary volume is randomly selected for each unique controller. All other eligible volumes behind the same controller are secondary. Secondary volumes are randomly selected if initial allocation on the primary volume is unsuccessful. There is no preference in different controller models as long as the controller supports striping.

SMS randomly selects a storage group from a list of storage groups that contain at least as many unique controllers as the requested stripe count. If there are no storage groups that meet this criterion, the storage group with the largest number of unique controllers is selected (randomly if there are more than one).

Volumes that meet the requested MSR are preferred over volumes that do not meet the requested performance. In DFSMS 1.1, a volume is considered to meet the requested performance if the volumes meets or exceeds the request. In DFSMS 1.2 and later, a volume meets the requested performance if the volume's performance is within a predetermined range of the requested MSR.

Rules for Striping Volume Selection

The key rules for striping volume selection are:

- Storage groups containing mixed device types are not considered.
- Eligible volumes must remain below the 'high threshold' defined in the storage group after the stripe is allocated. This is different from non-striped allocations where no volume is rejected because of space, although they can be given a lower priority.
- Quiesced and enabled volumes are treated identically for striped allocations, unlike non-striped allocations where enabled volumes are preferred over quiesced volumes.
- The number of target volumes is computed by dividing the SDR specified in the storage class by a value of 3 for 3380 devices, and by a value of 4 for 3390 devices, and rounding up the result if required. For example, an SDR of 18 results in a target stripe count of 6 on a 3380 device and a target stripe count of 5 (after rounding up) on a 3390 device. An SDR of 0 results in a target stripe count of 1, and conventional volume selection occurs. A non-zero SDR which is 1 when divided by the device value results in a target stripe count of 1, but striping volume selection is used instead of conventional volume selection.
- All temporary data sets with a volume count greater than one are allocated as non-striped.
- The volume must be able to satisfy the primary space requested by the number of stripes and not exceed the high threshold specified in the storage group.
- For guaranteed space allocations, the target number of stripes is the greater of either the volume count specified or the number of specified volume serial numbers. For a guaranteed space request, all specified volumes must be in the same SMS storage group. SMS assumes that the amount of space that the user wants is the target number of stripes times the primary space specified.
  – If you explicitly specify volume serial numbers with guaranteed space and the target number of stripes is equal to the number of volume serial numbers you specify, SMS must allocate the primary space requested on each of these volumes. If this is not possible, the allocation fails.
  – If you do not specify any volume serial numbers, then the target number of stripes is equal to the volume count. SMS tries to select the same number of volumes, but settles for less if this number is unavailable. If fewer stripes are allocated, SMS increases the allocation per volume to compensate for the fewer stripes.
  – If the target number of stripes is higher than the number of volume serial numbers you specify, SMS must select all the specified volumes plus additional ones. These non-specific volumes are not mandatory and if none are available, SMS allocates the primary quantity on each of the specific volumes.
  – It is recommended that you have a sufficient number of volumes behind each controller in the storage group to reduce volume overuse. An example of a problem would be if a storage group contains eight controllers and the average stripe count is four, each controller is selected approximately 50 percent of the time. If one controller contained only two volumes, each of the two volumes is selected for approximately 25 percent of all new striped allocations.

- The maximum number of stripes (volumes) is 16.
- The maximum number of extents is five for each space allocation.
- The maximum number of extents per stripe is 123.
- The minimum allocation is one track per stripe.
- The maximum allocation can exceed the 64K track limit and is limited by the amount of free space below the high threshold or extent constraints mentioned above for primary space allocations.
- All stripes must be able to satisfy the secondary space allocation (divided by the number of stripes) during extend processing or the allocation fails. Secondary space amount is divided by the number of stripes for both guaranteed and non-guaranteed space requests. OPEN, CLOSE, and EOV perform this.
- Multi-striped data sets cannot extend to additional volumes.
- Volume fragmentation information is not available to SMS at volume selection time. An allocation failure by DADSM because of fragmentation results in striping volume re-selection.
- Primary space
  - For non-guaranteed space, the volume must be able to satisfy the primary space requested divided by the number of stripes. For example, if primary space is 15MB, and the number of stripes is three, the volume must be able to satisfy an allocation of 5MB and not exceed this high threshold as specified in the storage group.
  - For guaranteed space, requests that contain specified volume serial numbers, each stripe must be able to satisfy the primary space requested (15MB in above example).
- Secondary space amount is divided by the number of stripes and rounded up for each volume.

Requirements for Data Set Striping

The following lists the requirements for data set striping.

- In DFSMS 1.1, volumes must be behind controllers that are ESCON-attached and support concurrent copy.
- In DFSMS 1.2 and later, volumes must be behind one of the following controllers:
  - Controllers that are ESCON-attached and support concurrent copy
  - 3990-6 controllers
  - 3990-3 controllers which are Extended Platform and ESCON-attached
  - 3990-3 controllers which have the RAMAC support-level microcode
  - 9394 controller
  - 9343 controller with cache
  - | IBM RAMAC Virtual Array
- Volumes must be ENABLED or QUIESCED and varied ONLINE.

Some Possible Reasons for Failure in Volume Selection

During the process of volume selection, volumes can be rejected for any one of many different reasons. Possible reasons for rejection include:

- The volume is not online to MVS.
- The SMS volume or storage group status is DISNEW, DISALL, or NOTCON.
- The volume was not initialized as being SMS-managed.

- The volume does not contain enough space to satisfy the primary space requested (this is determined by DADSM).
- The volume control data sets (VTOC) does not contain enough space to accomodate the FMT1 DSCB for the allocation request, as determined by DADSM.
- The volume was selected and DADSM returned an unsuccessful return/reason code for the allocation request. When DADSM fails to allocate the data set on a volume due to insufficient space on the volume, SMS tries to allocate the data set on a different volume. Other failures in DADSM can result in allocation failures.
- The volume does not meet the availability requirement specified in the storage class.
- The volume does not support concurrent copy and the storage class specifies accessibility continuous (Required).
- The DASD controller does not support extended format and the data class specifies 'If EXT=R.'
- The volume is mountable DASD when an IART of zero or blank was specified in the storage class.
- The volume was listed on the provided exclude list or was not listed in the provided include list.
- The volume has an incorrect unit control block (UCB) type.
- The volume was not specified for a specific guaranteed space request.
- The volume belongs to a storage group and the storage group does not contain enough volumes to satisfy the volume count of a guaranteed space request. All volumes in the storage group are rejected.
- The volume is reserved or enqueued.
- The volume is too fragmented to contain the primary space extent.
- During extend (EOV) processing, volume selection rejects volumes whose device types do not match the device type of the volume that the data set currently resides on.
- The controller was IMLed while online to MVS. This can result in the MVS device control blocks not reflecting the current state of the volume. The device or devices should be varied offline and back online to update the MVS control block status. SMS can reject the device because of incorrect MVS status. ISMF reports the correct status because ISMF queries the device/controller directly. Because of I/O performance, SMS uses the status in the MVS device control blocks for all volumes in a storage group.
- The volume is not assigned to any of the storage groups selected by the storage group ACS routine. This results in the volume not being included on the candidate volume list, which has the same effect as the volume being rejected.

Please note that during the process of volume selection many volumes can be rejected because they do not possess the right attributes (for example, allocation requires extended format but the volume does not support extended format). With few exceptions, volume selection fails only after allocation has been attempted without success on each of the remaining volumes.

Chapter 7. Defining Data Classes

When allocating data sets without SMS, end users need to specify many data set attributes. If you had many data sets with similar allocation requirements, you probably did a lot of copying or repeated typing. SMS simplifies the allocation of data sets by introducing data classes.

This chapter describes data classes and shows you how to define them using the ISMF data class application.

Understanding Data Classes

A data class is a list of data set allocation attributes and their values. You cannot assign a data class to an object. When end users allocate a data set and refer to a data class either explicitly (for example, through JCL) or implicitly (through ACS routines), SMS allocates the data set using the attribute values of its associated data class. For data class attributes, explicit specifications by end users override any parameters derived from the data class ACS routine. If SMS is active, a data class can be assigned to any data set. For data sets that are not SMS-managed, the system uses the allocation attribute values of the data class, but it does not save the data class name anywhere.

Not all attributes apply to every data set organization. When SMS allocates a data set, it uses only those data class attributes that have meaning for the given data set organization. SMS saves the data class name for each SMS-managed data set. The actual data class definitions reside in the SCDS. If you alter a data class definition, SMS applies the changes to any new data sets that use the data class after you activate the changed configuration. However, SMS does not retroactively apply your changes to previously allocated data sets.

A data class definition contains identification and allocation information. To identify and refer to your data classes, you assign each one a unique name that contains from one to eight alphanumeric characters. Each data class maintains an owner ID that identifies the storage administrator who originally created or last modified the data class. The owner ID can be viewed on the Data Class List panel. Also, each data class contains an optional 120 character description field for describing its contents.

The data class allocation attributes match the keywords that you use to allocate data sets. The attributes contain space-related, access-related, and organizational information that you typically find on JCL DD statements, TSO/E ALLOCATE commands, access method services DEFINE commands, dynamic allocation requests, and ISPF/PDF panels.

Planning Data Classes

You should create a data class based on service level agreements. For example, all data sets having a low-level qualifier of LIST, LISTING, OUTLIST, or LINKLIST probably belong to the same data class, because they are typical work data sets having similar allocation characteristics.

Before you actually define any data classes, gather information about the common types of data sets in your installation. You also need to determine if only the data class ACS routine can assign data classes to data sets, if end users can assign data classes to data sets, or if you want a combination of these two policies. If you intend to have only the data class ACS routine assign data classes, you need to develop methods to identify the data in your installation. However, if you allow only the data class ACS routine to assign data classes, you should be aware that users need to override some data class attributes. For example, you should probably not code one data class for each possible amount of space that users need. Finally, you need to identify the space requirements for some commonly used data sets.

By gathering useful installation information, you can relieve your end users from specifying all of the allocation attributes on allocation requests. They can use the data class that most closely matches their needs and explicitly change the few attributes that are unique to their data sets.

You can create a data class with no attributes specified and use it to handle system-managed data sets that are allocated (but never opened) without all DCB attributes being specified. This might be a common situation with batch systems where the DSORG, for example, is specified in the program but not all allocated data sets are used for every run of the batch stream. The unused data sets then occupy space on DASD and cannot be migrated by DFSMShsm since they do not have a specified DSORG. To solve this, check in your data class ACS routine for allocations that do not specify a DSORG or a data class and assign them the blank data class. This causes the DSORG for the data set to default to physical sequential (PS), so that DFSMShsm can migrate the data set.

Defining Data Class Attributes

You can use ISMF to define your data classes by selecting option 4, DATA CLASS, from the ISMF Primary Option Menu for Storage Administrators. Figure 44 illustrates the Data Class Application Selection panel.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                       DATA CLASS APPLICATION SELECTION
Command ===>


To perform Data Class Operations, Specify:
  CDS Name . . . . . . 'SMS.SCDS1.SCDS'
                               (1 to 44 character data set name or 'Active' )
  Data Class Name  . . DATAV     (For Data Class List, fully or partially
                                 specified or * for all)

Select one of the following options  :
  3  1. List    - Generate a list of Data Classes
     2. Display - Display a Data Class
     3. Define  - Define a Data Class
     4. Alter   - Alter a Data Class

If List Option is chosen,
   Enter "/" to select option      Respecify View Criteria
                                   Respecify Sort Criteria


Use ENTER to Perform Selection;
Use HELP Command for Help: Use END Command to Exit.
```

Figure 44. Defining a Data Class

Add a CDS Name and a Data Class Name on the panel and select option 3, Define. The CDS Name must be the name of an SCDS. ISMF primes the CDS

Name field with the name last used. The default is the quoted word 'ACTIVE', which represents the currently active configuration, but you cannot define or alter data classes to the 'ACTIVE' configuration.

In the Data Class Name field, you must specify the name of the data class that you are defining. ISMF primes the field with the last used name.

Defining Record and Space Attributes for Data Class

Select option 3, DEFINE, and press ENTER to see page 1 of the Data Class Define panel shown in Figure 45. You can leave any of the pages of the Data Class Define panel at any time without saving the data class by issuing the CANCEL command. The first page of the Data Class Define panel contains data class record and space

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                              DATA CLASS DEFINE                 Page 1 of 3
Command ===>

SCDS Name  . . : SMS.SCDS1.SCDS
Data Class Name: DATAV

To DEFINE Data Class, Specify:
  Description ==>
              ==>
  Recorg  . . . . . . . .              (KS, ES, RR, LS or blank)
  Recfm . . . . . . . . . VB           (any valid RECFM combination or blank)
  Lrecl . . . . . . . . . 255          (1 to 32761 or blank)
  Keylen  . . . . . . . .              (0 to 255 or blank)
  Keyoff  . . . . . . . .              (0 to 32760 or blank)
  Space Avgrec  . . . . . U            (U, K, M or blank)
        Avg Value . . . . 255          (0 to 65535 or blank)
        Primary . . . . . 5000         (0 to 999999 or blank)
        Secondary . . . . 5000         (0 to 999999 or blank)
        Directory . . . .              (0 to 999999 or blank)

Use ENTER to Perform Verification; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 45. Defining Data Class Record and Space Attributes

attributes. SCDS Name and Data Class Name are output fields that contain the SCDS and data class names you specified in the Data Class Application Selection panel. The Description field is an optional field of 120 characters in which you can describe the data class. All of the remaining fields are optional and have a default value of blanks.

A blank indicates that no value is assigned to a parameter, and end users need to specify the value explicitly if it is required and they intend to use the data class. Otherwise, their jobs might fail. When users override some data class attributes, they should ensure that the remaining attributes in the data class do not conflict with the explicitly specified attributes.

```
LRECL < KEYOFF
DSORG = PO and 0 directory blocks.
```

Note: If you do not specify a Recorg value for data sets with a data class, assigned either by JCL or ACS routine, the DSORG defaults to either physical sequential (PS) or partitioned organization (PO). Data class does not have a DSORG field. To specify a physical sequential data set, specify

RECFM. To specify partitioned organization, specify Recfm and Space values for directory blocks. For more information, see *DFSMS/MVS Using Data Sets*.

You can specify the following attributes on the first page of the Define Data Class panel:

Recorg

specifies the data set organization, and it resembles the RECORG DD attribute. If you specify a Recorg value, you cannot specify the Recfm attribute. A blank value specifies either a physical sequential or a partitioned organization (non-VSAM data set).

Table 5 summarizes which data class attributes apply to each record organization (RECORG).

Table 5. Applying Data Class Attributes to Record Organization (Recorg)

Attribute	Record Organization (RECORG)				
	Blank	KS	ES	RR	LS
LRECL	x	x	x	x	
RECFM	x				
KEYLEN	x	x			
KEYOFF	x	x			
VOLUME COUNT	x	x	x	x	x
SPACE	x	x	x	x	x
CISIZE	x	x	x	x	x
IMBED	x	x			
REPLICATE	x	x			
FREESPACE	x	x			
SHAREOPTIONS	x	x	x	x	x
Retpd or Expdt	x	x	x	x	x
DSNTYPE	x	x			
COMPACTION	x	x			

See the following for more information:

DFSMS/MVS Access Method Services for ICF

DFSMS/MVS Using Data Sets

OS/390 V2R6.0 MVS JCL Reference

Recfm

specifies the data set record format. You can specify one of the following, with the option of appending either an A (ISO/ANSI control character) or an M (Machine control character):

U Undefined

V Variable

VS Variable spanned

VB Variable blocked

VBS Variable blocked spanned

F Fixed

FS Fixed standard

FB Fixed blocked

FBS Fixed blocked standard

If you specify a Recfm value, you cannot specify the Recorg attribute.

LRECL
specifies the logical record length in bytes. If you leave the Recorg field blank, you can specify a Lrecl value from 1 to 32760 (or leave it blank). If you choose a Recorg value of KS, ES, or RR, you can specify a Lrecl value from 1 to 32761 (or leave it blank). If the Recorg value is KS, the Lrecl value must be greater than or equal to the value you specify for the Keylen attribute. Also if the Recorg value is KS and you specify values for both the Keylen and Keyoff attributes, then the Lrecl value must be greater than or equal to the sum of the Keylen and Keyoff attributes.

If the Recorg value is LS, the Lrecl attribute is ignored and the CIsize value is 4096.

Keylen
specifies the key length in bytes. To use this attribute, you must specify either KS or blank for the Recorg attribute. If the Recorg value is KS, the Keylen attribute represents the length of the KSDS key field and ranges from 1 to 255, or you can leave it blank. If the Recorg value is blank, the Keylen value ranges from 0 to 255, or you can leave it blank. For either value of Recorg, you must assign a value to the Keylen attribute that is less than or equal to the Lrecl value.

Keyoff
specifies the displacement from the beginning of a record to the KSDS key field. It is valid only when the Recorg value is KS. The Keyoff value can range from 0 to the value of (Lrecl - Keylen).

If a Keylen value is specified, then the Keyoff attribute must also be specified, or 0 value is used. This value is merged only if the Keylen value is not specified in JCL.

Space Specifies a request for space in records, eliminating track and cylinder space requests.

Avgrec
specifies a scaling factor for primary and secondary record allocations. It can have the following factors:

U Multiplies the allocation quantity by 1.

K Multiplies the allocation quantity by 1024.

M Multiplies the allocation quantity by 1048576.

Avg Value
specifies the average length of each record.

Primary
specifies the primary allocation quantity of records as multiplied by the scaling factor (Avgrec).

Secondary
specifies the secondary allocation quantity of records as multiplied by the scaling factor (Avgrec).

Directory
specifies the number of directory blocks for a PDS. It is valid only when the following fields are blank:

- Recorg
- Keyoff

- Imbed
- Replicate
- CIsize Data
- % Freespace CI
- % Freespace CA
- Shareoptions Xregion
- Shareoptions Xsystem

If you specify the following values:

Avgrec = K

Avg Value = 80

Primary = 500

Secondary = 100

Directory = 100

you request 40000KB of primary space, 8000KB of secondary space, and 100 directory blocks. The JCL used to accomplish the same task looks like:

```
//DD1  ...   SPACE=(80,(500,100,100)),AVGREC=K   ...
```

Defining Volume and VSAM Attributes for Data Class

From Page 1 of the Data Class Define panel, issue the DOWN command to view Page 2, shown in Figure 46.

The second page of the Data Class Define panel contains additional volume and

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                             DATA CLASS DEFINE           Page 2 of 3
Command ===>

SCDS Name . . . :
Data Class Name :
To DEFINE Data Class, Specify:
  Retpd or Expdt  . . . . . . __________  (0 to 9999, YYYY/MM/DD or blank)
  Volume Count  . . . . . . . 1           (1 to 59 or blank)
    Add'l Volume Amount . . . _           (P=Primary, S=Secondary or blank)
  Imbed . . . . . . . . . . . _           (Y, N or blank)
  Replicate . . . . . . . . . _           (Y, N or blank)
  CIsize Data . . . . . . . . _____       (1 to 32768 or blank)
  % Freespace CI  . . . . . . ___         (0 to 100 or blank)
              CA  . . . . . . ___         (0 to 100 or blank)
  Shareoptions Xregion  . . . _           (1 to 4 or blank)
               Xsystem  . . . _           (3, 4 or blank)
  Compaction  . . . . . . . . _           (Y, N, T, G or blank)
  Media Interchange
    Media Type  . . . . . . . _           (1, 2, 3, 4 or blank)
    Recording Technology  . . ___         (18, 36, 128 or blank)
  Use ENTER to Perform Verification; Use UP/DOWN Command to View other Panels;
  Use HELP Command for Help; Use END Command to Save and Exit; Cancel to Exit.
```

Figure 46. Defining Data Class Volume and VSAM Attributes

VSAM attributes for the data class. SCDS Name and Data Class Name are output fields that contain the SCDS and data class names you specified on the Data Class Application Selection panel. All of the remaining fields are optional.

Retpd or Expdt

specifies the retention period or expiration date for the data class being defined. Retention period is the number of days (0 to 9999) and expiration date is the date when you want the definition to expire. The default is 'blank', no expiration time.

Volume Count
specifies the maximum number of SMS-managed volumes that a data set can span. This attribute is ignored for data sets to which no storage class is assigned. The default is 1.

Add'l Volume Amount
specifies whether primary or secondary allocation amounts are to be used when the data set is extending to a new volume. You can specify:

- P for primary
- S for secondary

If you leave the field blank, the system default of primary is used. This attribute is used during VSAM EOV processing, and is only applicable to any VSAM multivolume data sets allocated in the extended format.

Imbed specifies whether the sequence set is to be written with the data component. If you specify Y, yes, the sequence set record for each control area is written on the first track adjacent to the control area, as many times as it fits. If you specify N, no, the sequence set record for each control area is written with the other index records. The Imbed attribute is valid only if the Recorg value is KS or blank. Note that the Imbed attribute is ignored when a KSDS is allocated in extended format. The default is blank.

Replicate
specifies whether each index record is to be written on a track as many times as it fits (Y) or only once (N). The Replicate attribute is valid only if RECORG is KS. The default is blank.

CIsize DATA
specifies the control interval size for the data component of data sets having Recorg values of KS, ES, RR,or LS. The default is 4096.

% Freespace
specifies the percentage of space to be left free when a cluster is loaded. It also specifies the percentage of space to be left free after any split of control intervals (CI) or control areas (CA). It applies only to the data component. The Freespace attribute is valid only if the Recorg value is KS or blank. The default is blank.

Shareoptions
specifies how end users can share a component or cluster.

The Xregion value specifies the amount of sharing allowed among regions within the same system, or within multiple systems using global resource serialization (GRS). You can specify the following values:

1 Any number of users can read the data set at one time, *or* only one can write to it. This setting does not allow any type of non-RLS access when the data set is already open for RLS processing.

2 Any number of users can read the data set at one time, *and* only one can write to it. If the data set is already open for RLS, non-RLS users can read the data set but cannot write to it. If the data set has been opened for non-RLS output, an RLS open fails.

Note: You must apply APARs OW25251 and OW25252 to allow non-RLS read access to data sets already open for RLS processing.

3 Any number of users can share the data set, and each is responsible for maintaining read and write integrity. This setting does not allow any type of non-RLS access when the data set is already open for RLS processing.

4 Any number of users can share the data set, and buffers used for direct processing are refreshed for each request. This setting does not allow any type of non-RLS access when the data set is already open for RLS processing.

The Xsystem value specifies the amount of sharing allowed among systems. You can specify the following values:

3 Any number of users can share the data set, and each is responsible for maintaining read and write integrity.

4 Any number of users can share the data set, and buffers used for direct processing are refreshed for each request.

If values are not specified explicitly or in the data class, VSAM defaults are used.

Compaction
specifies whether data is to be compressed. You can compress data on tape, or on DASD if the data set is allocated in the extended format. This field is ignored for DASD data sets if the data set name type is not EXT. A compressed data set cannot reside on the same cartridge as a data set that is not compressed.

For physical sequential data sets, you can specify T or G to compress using either tailored or generic compression dictionaries. This overrides the compression option set in PARMLIB, and lets you select the type of compression on a data set level. Current users of generic dictionaries can move to using tailored dictionaries a data set at a time, as new data sets are created.

Media Interchange
provides the capability to control the type of media created to make the data and media acceptable to other processors in the same or different locations. For example, you can create a data class with a name of BOSTON and define the media interchange parameters (Media Type and Recording Technology attributes) compatible with the tape hardware in the Boston location. When the hardware in Boston is changed, the BOSTON data class can be changed to the new media interchange parameters.

Media Type
specifies mountable tape media cartridge type. Valid values are MEDIA1, MEDIA2, MEDIA3, MEDIA4, or blank. This field is optional.

Recording Technology
specifies recording technology for mountable tape media cartridges. This field is dependent on the value specified in the Media Type field:

- If Media Type is MEDIA1, valid values are 18TRACK or 36TRACK.
- If Media Type is MEDIA2, only 36TRACK is valid.
- If Media Type is MEDIA3 or MEDIA4, only 128TRACK is valid.

This field is optional. If you leave this field blank, the system default is 36TRACK.

Defining Data Set Type Attributes for Data Class

From Page 2 of the Data Class Define panel, use the DOWN command to view Page 3, shown in Figure 47:

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                             DATA CLASS DEFINE            Page 3 of 3
Command ===>

SCDS Name . . . : USER2.TEST.ACDS
Data Class Name : ATLDC008

To DEFINE Data Class, Specify:
  Data Set Name Type  . . . . . . ___  (EXT, HFS, LIB, PDS or blank)
    If Ext  . . . . . . . . . . . _    (P=Preferred, R=Required or blank)
    Extended Addressability . . . N    (Y or N)
    Record Access Bias  . . . . . _    (S=System, U=User or blank)
  Reuse . . . . . . . . . . . . . N    (Y or N)
  Initial Load  . . . . . . . . . _    (S=Speed, R=Recovery or blank)
  Spanned / Nonspanned  . . . . . _    (S=Spanned, N=Nonspanned or blank)
  BWO . . . . . . . . . . . . . . __   (TC=TYPECICS, TI=TYPEIMS, NO or blank)
  Log . . . . . . . . . . . . . . _    (N=NONE, U=UNDO, A=ALL or blank)
  Logstream Id  . . . . . . . . . __________________________
  Space Constraint Relief . . . . N    (Y or N)
    Reduce Space Up To (%)  . . . __   (0 to 99 or blank)

Use ENTER to Perform Verification; Use UP Command to View previous Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit
```

Figure 47. Defining Data Class Type and Addressability

Page 3 allows you to specify the following:

- The type of data set being allocated.
- Additional attributes for data sets allocated in extended format, such as whether allocation in extended format is preferred or required, whether extended addressability is used, and whether to let VSAM determine how many and which type of buffers to use when allocating the data set.
- Whether the VSAM cluster can be reused.
- How the data set is to be loaded.
- Whether to support VSAM data sets with spanned record formats.
- Whether to assign attributes for VSAM record-level sharing (RLS) to system-managed data sets.
- Whether to retry new volume allocations or extends on new volumes that have failed due to space constraints.

Data Set Name Type

specifies the format in which the data set is to be allocated. When you specify extended format, you can also select requirements for the data set, including the need for extended addressability.

Partitioned data sets might reside in either SMS-managed or non-managed storage, but PDSEs must reside in DASD SMS-managed storage. You get an error if you allocate a data set in non-SMS-managed storage and its data class contains a data set name type of LIB.

For detailed information on assigning and allocating data classes for extended format data sets, see *DFSMS/MVS Using Data Sets*.

The available values are:

EXT specifies that the data set to be allocated is in the extended format. All VSAM data set types can be allocated in the extended format (with the exception of key range data sets, temporary data sets, and system data sets—including all catalogs).

If you specify EXT, you must also specify a value for the If Ext subparameter. The other subparameters are primed with the following defaults: Extended Addressability = No, and Record Access Bias = USER.

Note: You cannot use JCL, ALLOCATE, or DEFINE to specify EXT.

HFS specifies that the data set is a hierarchical file system (HFS) data set.

LIB specifies that the data set is a PDSE.

PDS specifies that the data set is a partitioned data set.

blank leaves the Data Set Name Type attribute value unspecified.

If you specify a Data Set Name Type of EXT for data sets allocated in extended format, you can specify additional attributes:

If Ext specifies whether allocation in extended format is preferred or required. If you specified EXT for the Data Set Name Type attribute, you must also specify one of the following:

P (preferred)
The data set allocation is attempted in non-extended format if the necessary system resources for extended are not available.

R (required)
The data set allocation fails if the data set cannot be allocated in extended format.

Extended Addressability
specifies that a VSAM data set in the extended format be allocated using extended addressability. This lets the data set grow beyond the four gigabyte (4GB) size. The data set can be a VSAM data set in any record organization but must be allocated in the extended format.

The available values are:

Y Extended addressability is used.

N Extended addressability is not used. This is the default.

Record Access Bias
specifies whether to let VSAM determine how many and which type of buffers to use when accessing VSAM extended format data sets by batch processing. This is known as system-managed buffering, and is available to VSAM data sets in any record organization which are allocated in the extended format. Note that the buffering algorithms determined by VSAM can be overridden via JCL.

The available values are:

System
specifies that VSAM chooses how many buffers to use and how they are processed.

System-managed buffering only takes effect if the application requests the use of non-shared resources (NSR). It is not effective for applications requesting local shared resources (LSR), global shared resources (GSR), or record-level sharing (RLS).

The number of buffers and the buffering technique are determined by:

- The system, based on application specifications (ACB MACRF=(DIR,SEQ,SKP)) and the values for direct and sequential millisecond response (MSR) and bias specified in the storage class.
- The user, as specified on the JCL DD card (AMP=('ACCBIAS=).

If sequential processing is to be used, the system optimizes the number of buffers and uses the NSR buffering technique. If direct processing is to be used, the system optimizes the number of buffers and uses the LSR buffering technique. You can change the defaults for space and the relative amount of hiperspace when direct optimization is used by using three new keywords in the AMP= parameter: VSMVSP=, SMBHWT=, and SMBDFR=.

User specifies that system-managed buffering is not used. This is the default.

Specifying Attributes for Data Set Reuse, Loading, and Spanned Record Format

On Page 3 of the Data Class Define panel, you can also specify attributes to indicate whether the VSAM cluster can be reused, how the data set is to be loaded, and whether to support VSAM data sets with spanned record formats.

Reuse specifies whether the VSAM cluster can be opened again as a new data set. Specify one of the following:

Y (Yes)
The cluster is reusable.

N (No)
The cluster is not reusable.

Initial Load
specifies how the data set is to be loaded. Specify one of the following:

S (Speed)
The data set is loaded without being preformatted.

R (Recovery)
The data set is preformatted when it is loaded. This is the default.

Spanned / Nonspanned
specifies whether a data record can span control interval boundaries. This applies to both system-managed and non-system-managed data sets. Specify one of the following:

Spanned
specifies that if the size of a data record is larger than a control

interval, the record can be contained on more than one control interval. This lets VSAM select a control interval size that is optimum for the DASD.

When a data record that is larger than a control interval is put into a cluster defined for spanned record format, the first part of the record fills a control interval. Subsequent control intervals are filled until the record is written into the cluster. Unused space in the record's last control interval is not available to contain other data records.

Note: Do not use this attribute for a variable-length relative record data set (VRRDS).

Nonspanned
specifies that a record must be contained in one control interval. VSAM selects a control interval size that accommodates the largest record in the data set. This is the default.

Specifying Attributes for Backup-While-Open (BWO) and Recovery

You can specify whether a data set is eligible for backup-while-open (BWO) processing. You can also indicate whether a data set is recoverable or not, and if so, provide the name of the forward recovery log stream.

Note that these attributes are only available to system-managed data sets.

BWO specifies whether BWO is to be used. This applies only to system-managed VSAM data sets, and is not available for linear data sets. Specify one of:

TYPECICS
BWO processing is used for CICS VSAM file control data sets. If you specify BWO(TYPECICS), CICS checks to see that you have also specified LOG(ALL) and that a logstream ID is available, either through the access method services DEFINE command or through the data class. Note that SMS rejects dynamic or JCL allocations if, as the result of an alter function, the value for LOG is changed to ALL without a logstream ID being available, or if the logstream ID is nullified.

TYPEIMS
BWO processing is used for IMS VSAM data sets.

Note: Support for this option is only available with IMS 6.1 and DFSMS/MVS 1.3 or above.

NO BWO is not used for CICS VSAM file control or IMS VSAM data sets. This is the default.

Log specifies whether the data set is considered recoverable or not.

Note that for data sets defined using access method services, the Log and Logstream ID attributes in the data class are merged with those defined in the DEFINE command. If a single logstream is to be used for each VSAM data set, the logstream ID should be specified on the DEFINE command. Specify the logstream ID in the data class only if the same logstream ID is to be used for many data sets. Otherwise, this will result in too many data classes.

Additionally, SMS rejects dynamic or JCL allocations if, as the result of an alter function, the value for LOG is changed to ALL without a logstream ID being available, or if the logstream ID is nullified. Specify one of:

NONE indicates that neither an external backout nor a forward recovery capability is available, so the data set is not considered recoverable.

UNDO indicates that changes can be backed out using an external log, so the data set is considered recoverable.

ALL indicates that changes can be backed out and forward recovered using an external log.

If you specify LOG(ALL), you must specify a logstream ID, either on the access method services DEFINE command or in the Logstream ID field in the data class.

blank The data set is not recoverable. This is the default.

Logstream Id

identifies the CICS forward recovery log stream. It applies to all components of the sphere. If you specify LOG(ALL), you must specify a logstream ID.

A logstream ID is made up of 1 to 26 characters, including separators. This name is made up of one or more segments, each containing one to eight alphabetic, numeric, or national characters. The first character of each segment must be an alphabetic or national character. Segments are joined by periods.

For data sets defined using access method services, the attributes in the data class are merged with those defined in the DEFINE command. If a single logstream is to be used for each VSAM data set, the logstream ID should be specified on the DEFINE command. Specify the logstream ID in the data class only if the same logstream ID is to be used for many data sets. Otherwise, this will result in too many data classes.

Specifying Attributes to Handle Space Constraints During Allocation

You can specify attributes on Page 3 of the Data Class Define panel to indicate whether or not to retry new data set allocations or extends on new volumes that fail due to space constraints.

During allocation, there might not be enough space on a volume to meet the requested space. SMS volume selection can sometimes solve this problem by trying all candidate volumes before failing the allocation. You can also use the Space Constraint Relief and Reduce Space Up To (%) attributes to request that an allocation be retried if it fails due to space constraints. SMS retries the allocation by combining any of the following:

- Spreading the requested quantity over multiple volumes
- Allocating a percentage of the requested quantity
- Using more than 5 extends

Space Constraint Relief

specifies whether or not to retry an allocation that was unsuccessful due to space constraints on the volume. Note that allocation is attempted on all candidate volumes before it is retried. This attribute applies only to system-managed data sets, and is limited to new data set allocations, and while extending the data set on new volumes. Specify one of the following:

Y specifies that SMS retry the allocation.

N specifies that SMS does not retry the allocation, so that allocation is not attempted on multiple volumes.

This is the default.

If you specify Y, SMS begins the retry process. This is a one- or two-step process, depending on the volume count you specified. For JCL allocations, SMS determines the volume count by taking the maximum of the unit, volume, or volser count. If these are not specified, SMS picks up a volume count from the data class. If there is no data class, SMS defaults the volume count to 1.

- If the volume count is 1 (one-step process)

  SMS retries the allocation after reducing the requested space quantity based on the Reduce Space Up To attribute. SMS simultaneously removes the 5-extent limit, so that SMS can use as many extents as the data set type allows.
- If the volume count is greater than 1 (two-step process)

  First, SMS uses a *best-fit* volume selection method to spread the primary quantity over more than one volume (up to the volume count). If this fails, SMS continues with the best fit method after reducing the primary quantity and removing the 5-extent limit.

  Note that you can remove the 5-extent limit without reducing the primary quantity by specifying 0 for the Reduce Space Up To (%) attribute.

For extends to new volumes, space constraint relief is strictly a one-step process. If regular volume selection has failed to allocate space, SMS reduces space or removes the 5-extent limit, but does not try the best-fit method.

The number of extents vary depending on data set type, as follows:

- Non-VSAM, non-extended format data sets: up to 16 extents on the volume
- Non-VSAM, extended format data sets, up to 123 extents.
- PDSE, up to 123 extents on the volume
- VSAM data sets, up to 255 extents per component but only up to 123 extents per volume per component

Reduce Space Up to (%)

specifies the amount by which you want to reduce the requested space quantity when the allocation is retried. You must specify Y for the Space Constraint Relief attribute. Valid values are 0 to 99.

If you request space constraint relief but do not specify a percentage value (either 0 or blank), SMS does not reduce the requested space quantity. This implies your application cannot tolerate a reduction in the space to be allocated, so you want to remove the 5-extent limit, thereby allowing SMS to use more than 5 extents.

When you request space constraint relief in one or more data classes, you might notice any of the following:

- Very large allocations might succeed if a sufficiently large volume count is specified in the data class or through JCL.
- Existing data sets might end up with less space than initially requested on extents.
- The space allocated for new data sets might be less than requested.
- The number of extents used during initial allocation might result in fewer extents being subsequently available. For example, if the primary space allocation uses 10 extents when allocating a physical sequential data set, then only 6 extents are left for allocation of the secondary quantity.
- X37 abends should occur less frequently.

Compatibility PTFs are required for all DFSMS/MVS releases below 1.4, and for all DFP releases so that lower-level systems can access all 255 extents of a VSAM component created on a DFSMS/MVS 1.4 system. Note that this access is for input only. Do not assign data classes with space constraint relief for VSAM data sets if there is a possibility they might be accessed for output on a lower-level system.

Assigning Data Classes

You can define an ACS routine to determine data classes, or end users can explicitly specify a data class name on the following:

- JCL DD statements
- TSO/E ALLOCATE commands
- Access method services ALLOCATE and DEFINE commands
- Dynamic allocation requests such as through ISPF/PDF data set allocation panels

Note that end users can explicitly specify data set attributes, and these specifications take precedence over the data set attributes assigned through the data class ACS routine.

Specifying Data Classes outside ACS Routines

The syntax for specifying a data class on a JCL statement is:

```
DATACLAS=data-class-name
```

The format for specifying a data class on a TSO/E command is:

```
DATACLAS(data-class-name)
```

The format for specifying a data class on an access method service command is:

```
DATACLAS(-)
```

Processing of Data Class Attributes in JCL

The order of precedence for data class attributes in JCL is as follows:

1. Explicit specifications
2. LIKE and REFDD keywords
3. Data class definitions (explicit or derived) in REFDD statement
4. Data class definitions in the referencing DD

A data class does not need to be a self-contained, complete data organization. You can partially define the data set attributes in the data class definition (as a base) and the user can explicitly specify the remaining attributes. However, the merging of all attributes according to the order specified above must result in a valid data organization. In the example below,

```
//DD1  DD    ...,DATACLAS=DC2,RECORG=ES,...
//DD2  DD    ...,DATACLAS=DC1,LRECL=180,REFDD=DD1,...
```

The LRECL of 180 in DD2 is used first, and then the RECORG of ES in DD1 is used, regardless of the values specified in either the DC1 or the DC2 data class. For the remaining attributes that are not explicitly specified on the DD statements, SMS uses the values defined in the data class definition of DC2 and then DC1.

In this next example, the attributes of the data set referenced by the LIKE keyword are used after all other explicit specification but before data class DC3 attributes:

```
//DD3  DD    ...,DATACLAS=DC3,LRECL=180,LIKE=SAMPLE.DATA,....
```

Here, an LRECL of 180 is used. Then, SMS uses the Data Set Control Block (DSCB) information from the SAMPLE.DATA data set. Finally, the remaining attribute values are drawn from the DC3 data class.

A final example illustrates the use of REFDD:

```
//DD4  DD   ...,DATACLAS=DC4,...
//DD5  DD   ...,DSN=DS1,REFDD=DD4,DATACLAS=DC5,LRECL=180,...
```

In this example, the REFDD keyword specifies that the explicit attributes on the DD4 JCL statement are to be used second since explicit attributes on the DD4 JCL statement are used after the explicitly specified attributes. Next, the attributes from data class DC4 referenced in DD4 should be used. The remaining attributes are taken from data class DC5.

Note that the LIKE and REFDD keywords are mutually exclusive.

For information on determining data classes through ACS routines, see “Chapter 9. Defining ACS Routines” on page 139. For information on access method services DEFINE defaults and data class defaults, see *DFSMS/MVS Access Method Services for ICF*.

Defining Additional Data Classes

You can copy existing data classes and modify them to create new ones by using the COPY line operator from the List panel, which is explained in “Copying SMS Classes, Storage Groups, and Aggregate Groups” on page 204.

Chapter 8. Defining Aggregate Groups

Individual SMS data sets in pool storage groups are backed up based on the backup attributes that are in a management class. Using ISMF's aggregate group application, you can group and back up data sets according to application or backup requirements. This chapter describes aggregate groups and shows you how to define them using the ISMF aggregate group application.

Understanding Aggregate Groups

An aggregate group is an SMS construct that uses control information and data set lists to define an application or other group. The data stored in the aggregate group is the principal input to DFSMShsm application backup and recovery.

An aggregate group consists of backup criteria and a group of data sets selected for backup by the storage administrator according to application or other requirements. For further information on backing up or recovering aggregate groups, see *DFSMS/MVS Managing Data Availability*. The aggregate group application allows you to:

- Generate a list of aggregate groups.
- Display the attributes of a single aggregate group.
- Define or alter the attributes of a single aggregate group.
- Delete aggregate groups.
- Edit and browse the selection data sets associated with aggregate groups.
- Edit and browse the instruction data sets associated with aggregate groups.
- Back up the selected aggregate group.
- Recover an aggregate group that has already been backed up.
- Specify one to fifteen local copies of aggregate backup (ABACKUP) output files to be created.
- Assign aggregate backup attributes to an aggregate group by specifying a management class name

Note: You cannot assign an object to an aggregate group.

ABACKUP allows specifying the number of copies of the Aggregate Backup and Recovery Support (ABARS) output files to be created by using the ISMF panel shown in Figure 49 on page 134. The Copies value can be 1 to 15, where 1 is the default.

In order to support the creation of multiple copies, a unique name must be created for each copy. This entails using a new suffix convention for the output data sets:

- .D.CccVnnnn for the DFSMSdss data file
- .O.CccVnnnn for the internal data file
- .C.CccVnnnn for the control file
- .I.CccVnnnn for the instruction or activity log file.

where

cc represents the copy number, which is a number from 1 to 15 of the copy created.

nnnn represents the version number.

The name supports the elimination of the GDG support.

Planning Aggregate Groups

First, you should identify those applications that are vital to continuing operation. Then, you must identify the application's associated components. These include:

- JCL and procedures
- Source, object, and load module data sets
- Procedures and run books
- Required system data sets
- Application data

Next, you should identify the required primary and recovery locations to ensure that the operating environments are compatible and the necessary resources are available. A primary location is the location that does an aggregate group backup of a specific application. A recovery location is the location where the aggregate group recovery of that specific application is performed.

Last, you should consider data set naming conventions to avoid duplicate data set names, and if system-managed data sets are to be recovered, you must ensure that the SMS constructs and attributes are compatible. Once the data sets needed to recover an application are identified, you can define the aggregate group.

Defining Aggregate Groups

You can use ISMF to define, alter, list, display, back up or recover an aggregate group, or edit the selection or instruction data sets associated with an aggregate group by selecting option 9, Aggregate Group, from the ISMF Primary Option Menu for Storage Administrators. A selection data set is a data set which contains lists of data sets to be included in the backup of an application or other group. An instruction data set is a data set which contains instructions, commands, and so on, that are copied into the control file volume after the backup control file.

For more information on Option 1, LIST, see "Listing SMS Classes, Aggregate Groups, Storage Groups, and Libraries Using ISMF" on page 190. Option 4, Alter, is discussed in "Altering Aggregate Groups" on page 201 and Option 5, Abackup and Option 6, Arecover, are discussed in "Backing up and Recovering an Aggregate Group" on page 137 and *DFSMS/MVS Managing Data Availability*.

Figure 48 on page 133 illustrates the Aggregate Group Application Selection panel.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                    AGGREGATE GROUP APPLICATION SELECTION
Command ===>


To Perform Aggregate Group Operations, Specify:
  CDS Name . . . . . . . . 'SMS.SCDS1.SCDS'
                                 (1 to 44 Character Data Set Name or 'Active')
  Aggregate Group Name . . TESTAG    (for Aggregate Group List, fully or
                                     Partially Specified or * for All)
Select one of the following Options:
  3  1. List     - Generate a list of Aggregate Groups
     2. Display  - Display an Aggregate Group
     3. Define   - Define an Aggregate Group
     4. Alter    - Alter an Aggregate Group
     5. Abackup  - Backup an Aggregate Group
     6. Arecover - Recover an Aggregate Group

If List Option is Chosen,
  Enter "/" to select option      Respecify View Criteria
                                  Respecify Sort Criteria

Use ENTER to Perform Selection;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 48. Aggregate Group Application Selection

To define an aggregate group, you must specify a CDS Name and an Aggregate Group Name on the panel and select option 3, Define. The CDS Name must be the name of an SCDS. ISMF primes the CDS Name field with the name last used for an aggregate group. The default is 'Active,' which represents the currently active configuration, but you cannot define or alter aggregate groups to the 'Active' configuration.

In Aggregate Group Name, you must specify the name of the aggregate group that you are defining. ISMF primes the field with the name last used.

Defining Aggregate Group Attributes

Select option 3, Define, and press ENTER to see the first of the two pages of the Aggregate Group Define panel shown in Figure 49 on page 134. You can leave either page of the Aggregate Group Define panel at any time without saving the aggregate group attributes or changes by issuing the CANCEL command.

Note: Three aggregate group attributes have been removed in DFSMS/MVS. They are Expiration Date, Destination, and Tolerate Enqueue Failure. Aggregate groups defined in prior releases are accepted in DFSMS/MVS and vice versa. The three deleted attributes remain in aggregate groups defined prior to DFSMS/MVS, but have no effect. Refer to “Processing Aggregate Groups” on page 194 for more information on processing aggregate groups.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                           AGGREGATE GROUP DEFINE              Page 1 of 2
Command ===>

SCDS Name  . . . . . : SMS.SCDS1.SCDS
Aggregate Group Name : TESTAG

To DEFINE Aggregate Group, Specify:
 Description ==>
             ==>

 Backup Attributes
   Number of Copies  . . . . . . . 1                (1 to 15)
   Management Class Name . . . . .                  (1 to 8 characters, to be
                                                    defined in current SCDS)

 Output Data Set Name Prefix . . .
                                                    (1 to 33 Characters)
 Account . . . . . . . . . . . . .
                                                    (1 to 32 Characters)

Use ENTER to Perform Verification; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 49. Aggregate Group Define, Page 1 of 2

Page one of the Aggregate Group Define panel contains aggregate group define attributes. The SCDS Name and Aggregate Group Name fields are output fields that contain the SCDS and aggregate group names you specified on the Aggregate Group Application Selection panel. The Description field is an optional field of 120 characters in which you can describe the aggregate group.

You can specify the following *required* attributes on the first page of the Aggregate Group Define panel:

Number of Copies
: specifies the number of aggregate backup output files to be created. The valid values are 1 to 15.

Management Class Name
: specifies the management class name from which the Aggregate Backup attributes are obtained. The valid values are 1 to 8 alphanumeric characters (first character not a digit) or a blank.

Output Data Set Name Prefix
: identifies the output data sets created by aggregate backup.

Editing Aggregate Group Attributes

After specifying your backup attribute values, issue the DOWN command to view the page two of the Aggregate Group Define panel, which is shown in Figure 50 on page 135.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                           AGGREGATE GROUP DEFINE              Page 2 of 2
Command ===>


SCDS Name  . . . . . : SMS.SCDS1.SCDS
Aggregate Group Name : TESTAG
To Edit a Data Set, Specify Number  . .     (1, 2, 3, 4, 5, or 6)
  Selection Data Sets:            (1 to 44 characters)
   1 ===>
     Member Name ===>             (1 to 8 characters)
   2 ===>
     Member Name ===>             (1 to 8 characters)
   3 ===>
     Member Name ===>             (1 to 8 characters)
   4 ===>
     Member Name ===>             (1 to 8 characters)
   5 ===>
     Member Name ===>             (1 to 8 characters)

  Instruction Data Set:           (1 to 44 characters)
   6 ===>
Use ENTER to Perform Verification; Use UP Command to View previous Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 50. Aggregate Group Define, Page 2 of 2

Page two of the Aggregate Group Define panel contains the selection and instruction data set names for the aggregate group. the SCDS Name and Aggregate Group Name fields are output fields that contain the SCDS and aggregate group names you specified on the Aggregate Group Application Selection panel.

Edit a Data Set
: Select the number of a selection or instruction data set that you want to edit. When you select a data set number, it allows you to allocate or modify the data set by invoking PDF Edit. The PDF edit screen is shown in Figure 51 on page 136. See *OS/390 ISPF User's Guide* for more information on the PDF Edit commands.

Selection Data Set Name
: Name of the data set containing lists of data sets to be included in the application backup. You can specify up to five selection data set names. One data set name is required. There is no default. If you want to enter a fully qualified data set name, enclose the name in single quotes. If you do not enclose the name in single quotes, the TSO prefix is added to the name as the first high level qualifier. There is no default.

Member Name
: Name of the data set member containing lists of data sets to be included in the application backup if the selection data set is partitioned. This name must be a valid TSO data set member name. Enter a valid member of the partitioned data set specified in the Selection Data Set Name field. This is required if the data set specified in the Selection Data Set Name field is a partitioned data set. If you want to enter a fully qualified data set name, enclose the name in single quotes. If you do not enclose the name in single quotes, the TSO prefix is added to the name as the first high level qualifier. There is no default.

Instruction Data Set Name
: Name of the data set containing instruction, commands, etc., that are copied into the control file volume after the backup control file. This data set

can only be a sequential data set. You must use a valid TSO data set name. The data set name, including the TSO prefix, can be no more than 44 characters long. This is an optional field and has no default.

If you select the option to edit a Selection or Instruction Data Set on the Aggregate Group Define panel, you get the PDF Edit screen shown in Figure 51.
You can specify the following keywords with parameters using PDF edit:

```
EDIT --- SELECT.DATASET.ONE ------------------------- LINE 000000 COL 001 072
COMMAND ===>                                                 SCROLL ===> HALF
******  **************************** TOP OF DATA ******************************
000001 INCLUDE( XMP.** )
000002 EXCLUDE( XMP.USER.TAPE1,
000003          XMP.INPUT.MASTER )
000004 ACCOMPANY( DATA.MASTER,          /* MASTER */
000005            XMP.USER.TAPE1 )
000006 ALLOCATE( XMP.OLD.DASD1 )
****** *************************** BOTTOM OF DATA *****************************
```

Figure 51. Editing an Aggregate Group Selection Data Set- Example

INCLUDE
specifies which data sets are included in the backup. No distinction is made whether they are tape or DASD. If a partitioned data set is indicated, all members are included in the application backup.

EXCLUDE
specifies the data sets that are specifically excluded from the backup process.

ACCOMPANY
specifies data sets that are physically removed from the backup site, transported to the recovery site, and only need to be cataloged during application recovery.

ALLOCATE
allocates and catalogs data sets at the recovery site.

When you allocate or edit selection data sets, there are several rules you must follow:

- Records must be 80 bytes in length and of a fixed format.
- Entries must be entered between columns 1 and 72.
- INCLUDE, EXCLUDE, ACCOMPANY, and ALLOCATE can be specified only once.
- A comment is a string of characters preceded by “/*” and followed by “*/” and might span multiple records.
- A separator consists of a comma (,), one or more blanks, or a comment.
- Parameters are separated from one another by one or more separators.
- One or more blanks can, optionally, precede and follow each parenthesis in the pair.
- Continuation of a record is optionally specified by a hyphen (-) or plus sign (+) as the rightmost non-blank character, preceded by one or more blanks. Continuation characters are not required.
- If the same fully qualified data set name is specified in both the INCLUDE list and the EXCLUDE list, the data set is not selected for processing because EXCLUDE takes precedence over INCLUDE.

- Data set names specified in the INCLUDE list cannot be specified in the ALLOCATE or ACCOMPANY list.
- The minimum truncation allowed for keywords is: **I** for INCLUDE, **E** for EXCLUDE, **AC** for ACCOMPANY, and **AL** for LOCATE.

Backing up and Recovering an Aggregate Group

To back up an aggregate group, select option 5, ABACKUP, on the Aggregate Group Application Selection panel, or enter the ABACKUP line operator on the Aggregate Group List panel, and press ENTER. You get the Aggregate Group Backup panel.

To recover an aggregate group, select option 6, ARECOVER, and press ENTER. You get the Aggregate Group Recover panel.

DFSMShsm handles your request for backup or recovery. You are prompted to enter information for backing up or recovering aggregate groups.

Defining Additional Aggregate Groups

You can copy existing aggregate groups and modify them to create new aggregate groups by using the COPY line operator, which is explained in “Copying SMS Classes, Storage Groups, and Aggregate Groups” on page 204.

Chapter 9. Defining ACS Routines

This chapter provides General-use Programming Interface and Associated Guidance Information.

This chapter is intended to help you to define ACS routines. ACS routines can be used to determine the SMS classes and storage groups for data sets and objects in an SMS complex. For storage administrators, ACS routines automate and centralize the process of determining SMS classes and storage groups. ACS routines also help convert data sets to an SMS environment.

An object is assigned to a storage group when it is stored and remains in that storage group throughout its lifetime. Initial storage class and management class might be determined by defaults defined by an ACS routine, or by explicit request which might be overridden by the ACS routines. Storage class and management class assignments might be changed by the OSREQ CHANGE function, or by automatic class transition. The OSREQ CHANGE request causes invocation of ACS routines that might override the requested assignments. During automatic class transition, ACS routines are invoked to determine the new storage class and management class assignments.

This chapter explains how to write ACS routines for an SMS configuration using the ISMF Automatic Class Selection Application.

“Chapter 15. ACS Language Reference” on page 257 lists the rules for programming in the ACS language.

Note: You can use the DFSMS/MVS NaviQuest tool to help you design and test your ACS routines. First, you can create test cases to perform extensive testing against test data representing actual data sets. Then you can test ACS routines in batch, freeing the workstation for other work. For more information on DFSMS/MVS NaviQuest, see *NaviQuest User’s Guide*.

Understanding ACS Routines

Through ISMF, you can create and maintain as many as four ACS routines in an SCDS, one for each type of SMS class and one for storage groups. After you have activated an SMS configuration, SMS executes ACS routines for the following operations:

- JCL DD statements (DISP=NEW, DISP=MOD)
- Dynamic allocation requests (DISP=NEW, DISP=MOD for a non-existent data set)
- DFSMSdss COPY, RESTORE, and CONVERTV commands
- DFSMShsm RECALL and RECOVER
- Access method services ALLOCATE, DEFINE, and IMPORT commands
- OAM processing for STORE, CHANGE, and class transition
- Local data set creation by remote application through Distributed FileManager/MVS
- MVS data sets or OS/390 UNIX System Service (OS/390 UNIX) files created by remote application through the DFSMS/MVS Network File System server.

As a storage administrator, you write ACS routines using the ACS programming language, a high-level programming language. The language follows a logical, procedural flow of implementation that consists mainly of filtering criteria, IF/THEN statements, and SELECT/WHEN statements. Using these relational statements, ACS routines determine SMS classes and storage groups according to allocation parameters, data set sizes, object or data set names, and other variables.

All allocations directed to units that are neither tape nor DASD should be excluded from SMS management. Do this by testing for UNIT in the storage class routine and ensuring that the storage class is set to NULL in these cases.

Ensuring that no storage class is assigned for such allocations avoids potential errors with allocations that require specific types of units. For example, assigning a storage class to a VTAM channel-to-channel (CTC) adaptor allocation results in sense errors when VTAM attempts to use the CTC.

Note: For system-managed data sets, the storage group is required because there is no way to explicitly specify storage groups. The other routines are optional. For objects, the storage group, storage class and management class ACS routines are required. For tape, the storage group, storage class and data class ACS routines are required.

Using ACS Routines for Data Sets Created by DFSMS/MVS Network File System and Distributed FileManager/MVS

ACS routines can be used to determine the SMS classes for MVS data sets and OS/390 UNIX files created by DFSMS/MVS Network File System. When data sets and files are created by DFSMS/MVS Network File System, any attributes related to SMS classes not specified by remote client user are defaulted using the ACS routines for the data set. If the remote user does not specify a storage class and if the ACS routines decide that the data set should not be SMS-managed, Distributed FileManager/MVSends and returns an error to the remote workstation. Therefore, it is important that on the MVS target system, you consider the potential data set creation requests of remote users when constructing the ACS routines.

When data sets are created by Distributed FileManager/MVS, any attributes not specified by the source requester are defaulted using the attributes specified in the data and management classes selected by the ACS routines for the data set. If the system requires attributes which neither the remote user nor the data class specified, the creation process fails.

For using ACS routines to determine the SMS classes for data sets created by Distributed FileManager/MVS, see “Determining Distributed FileManager/MVS Data Set Creation Requests” on page 275.

Restrictions on Using ACS Routines

ACS routines are not invoked for of data sets that cannot be system-managed.

- Dummy data sets
- Data sets that have SUBSYS coded on the JCL DD statement
- SYSIN and SYSOUT data sets

Creating ACS Routines

You must allocate either a sequential data set, a member of a partitioned data set (PDS), or a member of a partitioned data set extended (PDSE) for each of the ACS routines that you intend to write. As a general guideline, use an LRECL of 80. The maximum LRECL you can use is the maximum that can be specified on JCL. After allocating the data sets, select option 7, Automatic Class Selection Application, from the ISMF Primary Option Menu and press ENTER.

ISMF displays the Class Selection Application Selection panel shown in Figure 52.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                          ACS APPLICATION SELECTION
Command ===>


Select one of the following options:
1  1. Edit           - Edit ACS Routine source code
   2. Translate      - Translate ACS Routines to ACS Object Form
   3. Validate       - Validate ACS Routines Against Storage Constructs
   4. Test           - Define/Alter Test Cases and Test ACS Routines
   5. Display        - Display ACS Object Information
   6. Delete         - Delete an ACS Object from a Source Control Data Set



If Display Option is Chosen, Specify:

  CDS Name  . . 'SMS.SCDS1.SCDS'
                               (1 to 44 Character Data Set Name or 'Active')



Use ENTER to Perform Selection;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 52. Writing an ACS Routine

Select option 1, Edit, and press ENTER to write an ACS routine.

ISPF/PDF displays the Edit Entry panel shown in Figure 53 on page 142.

```
  Menu  RefList  RefMode  Utilities  LMF  Workstation  Help
------------------------------------------------------------------------------
                              Edit Entry Panel
Command ===>


ISPF Library:
   Project . . .
   Group . . . .          . . .          . . .          . . .
   Type  . . . .
   Member  . . .                 (Blank or pattern for member selection list)

Other Partitioned or Sequential Data Set:
   Data Set Name . . .
   Volume Serial . . .           (If not cataloged)

Workstation File:
   File Name . . . . .

Initial Macro  . . . .                  Enter "/" to select option
Profile Name . . . . .                  /  Confirm Cancel/Move/Replace
Format Name  . . . . .                     Mixed Mode
Data Set Password  . .                     Edit on Workstation
```

Figure 53. Invoking the ISPF/PDF Editor

After specifying your values, press ENTER to invoke the ISPF/PDF Editor. Then enter the source code for the ACS routine. After entering the routine, issue the END command to save it and return to the ISPF/PDF Edit Entry panel. Use END again to return to the Automatic Class Selection Application Selection panel.

Translating ACS Routines

After creating an ACS routine, you must translate it into executable form. The translation process checks your source code for syntactic and semantic errors, generates an object form if no errors exist, and places the object form into the SCDS you specified on the translate panel. If the ACS routine that you are translating already exists in the SCDS, the new object form replaces the existing object form.

To translate an ACS routine, select option 2, Translate, from the Automatic Class Selection Application Selection panel. ISMF displays the Translate ACS Routines panel shown in Figure 54 on page 143.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                            TRANSLATE ACS ROUTINES
Command ===>


To Perform ACS Translation, Specify:

   SCDS Name . . . . . . . 'SMS.SCDS1.SCDS'
                                     (1 to 44 Character Data Set Name)

   ACS Source Data Set . . 'SCDS1.STORGRP'
                                     (1 to 44 Character Data Set Name)

   ACS Source Member . . .           (1 to 8 characters)


   Listing Data Set  . . . 'STORGRP.LISTING'
                                     (1 to 44 Character Data Set Name)




Use ENTER to Perform ACS Translation;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 54. Translating an ACS Routine

A successful translation places the ACS routine object table into the SCDS that you specify in SCDS Name. SCDS Name is a required field that is primed with the last SCDS that you have referenced or translated into.

Specify the name of the data set containing the ACS routine that you want to translate in ACS Source Data Set. This is a required field that is primed with the last used value.

ACS Source Member is the name of the member of the source data set that contains the ACS routine. The name is required only if the ACS source data set is a partitioned data set or PDSE. The field is primed with the last used value. The default is blanks.

Specify the name of a sequential data set to contain the translation results in Listing Data Set. If you specify a data set that already exists, the translation process replaces the existing data set contents with the results of the translation. If you specify a non-existent data set, the translation process allocates space for it. If you leave this field blank, which is the default, you receive the results of the validation but you do not get a listing.

After specifying your values, press ENTER to perform the translation.

Browsing the Results of a Translation

If you specify an ACS Source Data Set that contains no errors, you get a PDF Browse panel similar to the one shown in Figure 55 on page 144 after translation.

```
 BROWSE -- STORGRP.LISTING--------------------------- LINE 00000000 COL 001 080
 COMMAND ===>                                                  SCROLL ===> PAGE
********************************* TOP OF DATA  *********************************
ACS TRANSLATOR ***** TIME 09:45:32 DATE 11/19/1987 PAGE 0001 *****

SCDS NAME:              'SMS.SCDS1.SCDS'

ACS SOURCE DATA SET:     SCDS1.STORGRP
ACS SOURCE MEMBER:

0001      PROC STORGRP
0002        SELECT
0003          WHEN(&STORCLAS = DB*)
0004            SET &STORGRP = 'DATABASE'
0005          WHEN(&MAXSIZE > 9999999KB)
0006            SET &STORGRP = 'LARGE'
0007          WHEN(&DSTYPE = 'TEMP')
0008            SET &STORGRP = 'VIO','PRIMARY'
0009          OTHERWISE
0010            SET &STORGRP = 'PRIMARY'
0011        END
0012      END

TRANSLATION RETURN CODE:  0000
******************************** BOTTOM OF DATA *********************************
```

Figure 55. Browsing the Results of an ACS Translation

The Browse panel displays the SCDS name, the ACS routine source data set name, an output listing of the ACS routine, and a translation return code. SCDS Name, ACS Source Data Set, and ACS Source Member reflect the values that you specified on the Translate ACS Routines panel. The output listing contains the source code of the ACS routine and diagnostic messages. See *MVS/ESA System Messages, Volumes 1–5* for an explanation of the diagnostic messages.

The Translation Return Code displays one of the following codes:

0000 Successful translation.

0012 Unsuccessful translation. The ACS routine contains one or more semantic or syntactic errors. No object form was created and no updates were made to the SCDS.

0020 Internal error in translator.

When you exit from this panel, you get the Output Listing Disposition panel shown in Figure 56 on page 145.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                          OUTPUT LISTING DISPOSITION
Command ===>



Listing Data Set : STORGRP.LISTING


Specify Output Listing Disposition:

  Enter "/" to select option   _  Print Output Listing
                               _  Delete Output Listing








Use ENTER to Perform Selection;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 56. Generating an ACS Translation Listing

The short message area displays the results of the translation. Specify Y in Print Output Listing to submit a batch job to print the data set. Specify Y in Delete Output Listing to delete the data set after it is printed. Both fields are required, and they are primed with N. Table 6 summarizes the possible outcomes for each combination of Output Listing Disposition values.

Table 6. Results of Output Listing Disposition

Print	**Delete**	**Resulting Action**	**Type of Job**
Yes	Yes	Print data set DISP=(OLD,DELETE)	Batch
Yes	No	Print data set DISP=(OLD,KEEP)	Batch
No	Yes	Delete data set	Foreground
No	No	No action (dataset kept)	N/A

Issue the END command to return to the Translate ACS Routines panel.

BROWSING the Results of an Unsuccessful Translation

If you specified an ACS source data set that contains errors, and you specified a data set in the Listing Data Set field, the Browse panel shown in Figure 57 on page 146 is displayed after translation.

```
 BROWSE -- STORGRP.LISTING--------------------------- LINE 00000000 COL 001 080
 COMMAND ===>                                                  SCROLL ===> PAGE
********************************* TOP OF DATA  *********************************
ACS TRANSLATOR ***** TIME 11:47:09 DATE 07/15/1991 PAGE 0001 *****

SCDS NAME:               'SMS.TEST.SCDS'
ACS SOURCE DATA SET:     SMS.ACS.ROUTINES
ACS SOURCE MEMBER:       SG1

0001      PROC STORGRP
0002        SELECT
0003          WHEN (&STORCLAS = DB*)
0004            SET &STORGRP = 'DATABASE'
0005          WHEN (&MAXSIZE > 99999999KB)
0006            SET &STORGRP = 'LARGE'
0007          WHEN &DSTYPE = 'TEMP'
*******   IGD03114I A LEFT PARENTHESIS WAS EXPECTED BUT NOT FOUND
0008            SET &STORGRP = 'VIO','PRIMARY'
0009          OTHERWISE
0010            SET &STORGRP = 'PRIMARY'
0011        END
0012      END

SUMMARY OF ERROR MESSAGES:

0007      IGD03114I A LEFT PARENTHESIS WAS EXPECTED BUT NOT FOUND

TRANSLATION RETURN CODE:  0012
**************************** BOTTOM OF DATA ****************************
```

Figure 57. Browsing the Results of an Unsuccessful ACS Translation

Scroll down to see a display of all the error messages together.

Validating ACS Routines or an Entire SCDS

SMS validates the entire SCDS when it is saved. You can validate the individual ACS routines of an SMS configuration after successfully translating them. You should also validate the entire configuration yourself so that you can see any error messages that result. Separate validation of the ACS routines does not produce all of the possible messages.

Validating an ACS Routine

To validate an ACS routine, select option 3, Validate on the Automatic Class Selection Application Selection panel. ISMF displays the Validate ACS Routines or Entire SCDS panel shown in Figure 58 on page 147.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                     VALIDATE ACS ROUTINES OR ENTIRE SCDS
Command ===>


To Perform Validation, Specify:

  SCDS Name  . . . . . 'SMS.SCDS1.SCDS'
                           (1 to 44 Character Data Set Name)

  ACS Routine Type . . SG  (DC=Data Class, MC=Management Class, SC=Storage
                           Class, SG=Storage Group, *=Entire SCDS)

  Listing Data Set . . SGVAL.LISTING
                           (1 to 44 Character Data Set Name)







Use ENTER to Perform Validation;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 58. Validating an ACS Routine

SCDS Name identifies the SCDS containing the ACS routine you are validating. It is a required field that is primed with the last used value. The default is blanks.

ACS Routine Type identifies which of the four ACS routine types you are validating. When validating an individual ACS routine, the valid values are: DC, SC, MC, and SG. It is a required field that is primed with the last used value. The default is an asterisk, *, which specifies that you want to validate the entire SMS configuration in the SCDS, and not just one of the ACS routines.

Listing Data Set identifies the name of a sequential data set to contain the results of the validation. If you specify a data set that already exists, the validation process replaces the existing data set contents with the results of the validation. If you specify a non-existent data set, the validation process allocates space for it. If you leave this field blank, which is the default, you receive the result of the validation, but you do not get a listing.

After specifying your values, press ENTER to perform the validation. Validation fails if the following conditions are not satisfied (except where it is noted that only a warning is issued):

- **For the storage group ACS routine**:

  All of the defined pool storage groups should be possible outcomes of the routine.

  No dummy storage group can result from the storage group ACS routine.

  All storage groups that are possible outcomes of the storage group ACS routine must exist (be defined).

  All VIO storage groups are possible results of the storage group ACS routine.

  Notes:

  1. If you have no pool storage groups, you get a warning.
  2. If you have any pool type storage groups that are not possible outcomes of the storage groups ACS routine (that is, are not used in any SET statement),

you get a warning. This is because the storage group cannot be explicitly requested and therefore represents a wasted resource.

- **For the management class ACS routine**:

  All management classes that are possible outcomes of the management class ACS routine must exist.
- **For the storage class ACS routine**:

  All storage classes that are possible outcomes of the storage class ACS routine must exist.
- **For the data class ACS routine**:

  All data classes that are possible outcomes of the data class ACS routine must exist.

If an ACS routine uses the SET statement to assign an SMS class or storage group that does not exist, validation fails. If an ACS routine references an SMS storage class, management class, or data class that does not exist, SMS issues a warning. For example, you might delete a storage class. If you continue to reference the deleted storage class in the storage class ACS routine of an SCDS, SMS issues a warning when you validate the SCDS. If only warnings exist, the SCDS becomes valid despite the warning. This allows you to check for the deleted storage class and replace it with an existing storage class within the storage class ACS routine. For example:

```
WHEN (&STORCLAS = 'OLDSC')
  SET &STORCLAS = 'NEWSC'
```

Figure 59. Checking for Deleted Storage Classes

BROWSING the Results of a Validation

If you specified a data set name in the Listing Data Set field, you get the Browse panel shown in Figure 60 after translation.
VALIDATION RESULT can be VALIDATION SUCCESSFUL, ERRORS DETECTED

```
 BROWSE -- SGVAL.LISTING ---------------------------- LINE 00000000 COL 001 080
 COMMAND ===>                                                  SCROLL ===> PAGE
********************************* TOP OF DATA  *********************************
                             VALIDATION RESULTS

VALIDATION RESULT:  ERRORS DETECTED

SCDS NAME:          'SMS.SCDS1.SCDS'

ACS ROUTINE TYPE:   SG
DATE OF VALIDATION: 1987/11/19
TIME OF VALIDATION: 10:01

IGD06025I THE STORAGE GROUP ACS ROUTINE SETS NON-EXISTENT VALUE DATABASE
IGD06025I THE STORAGE GROUP ACS ROUTINE SETS NON-EXISTENT VALUE LARGE
IGD06025I THE STORAGE GROUP ACS ROUTINE SETS NON-EXISTENT VALUE VIO
IGD06025I THE STORAGE GROUP ACS ROUTINE SETS NON-EXISTENT VALUE PRIMARY
IGD06025I THE STORAGE GROUP ACS ROUTINE SETS NON-EXISTENT VALUE PRIMARY
******************************** BOTTOM OF DATA ********************************
```

Figure 60. Browsing the Results of an ACS Routine Validation

or WARNINGS DETECTED. The SCDS Name and ACS Routine Type fields reflect the values you specified on the Validate ACS Routines or Entire SCDS panel.

Browse also provides the date and time of validation. See *MVS/ESA System Messages, Volumes 1–5* for an explanation of the diagnostic messages.

When you leave Browse, you receive the Output Listing Disposition panel. See Figure 56 on page 145 and Table 6 on page 145 for information about printing and deleting.

Validating an Entire SCDS

For its contents to become the active storage management policy for an installation, an SCDS must be valid. Activating an SCDS validates its contents and copies them into the ACDS identified by IGDSMS*xx*. If the SCDS is not valid, activation fails.

You can validate an entire SCDS using the ISMF VALIDATE command. To use the command, type `VALIDATE` on the command line of the CDS Application Selection panel (shown in Figure 7 on page 28) and press ENTER. You receive the Validate ACS Routines or Entire SCDS panel shown in Figure 61.
The SCDS Name field identifies the data set you want to validate. It is a required

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                     VALIDATE ACS ROUTINES OR ENTIRE SCDS
Command ===>


To Perform Validation, Specify:

  SCDS Name  . . . . . 'SMS.SCDS1.SCDS'
                             (1 to 44 Character Data Set Name)

  ACS Routine Type . . *   (DC=Data Class, MC=Management Class, SC=Storage
                             Class, SG=Storage Group, *=Entire SCDS)

  Listing Data Set . .
                             (1 to 44 Character Data Set Name)






Use ENTER to Perform Validation;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 61. Validating an SCDS

field that is primed with the last SCDS you have referenced or validated. You should place an asterisk (*) in the ACS Routine Type field, since you are validating an entire SCDS. You can save a listing of the validation results by specifying a sequential data set name in the Listing Data Set field. If you specify a data set that already exists, the old copy is replaced with your new listing. If you specify a non-existent data set, ISMF allocates space for it. If you leave this field blank, you do not generate a listing.

To define a valid minimal SMS configuration, the configuration must contain:

- A fully defined base configuration
- A storage class definition
- A pool storage group containing at least one volume, or a VIO, object, or tape storage group

- A storage class ACS routine
- A storage group ACS routine.

For more information on defining a minimal SMS configuration, see *DFSMS/MVS Implementing System-Managed Storage*.

Validation fails if any of the following conditions are not satisfied (except where it is noted that only a warning is issued).

- Base configuration information must exist. If you specify a default management class, you need to have defined the management class.
- Each possible outcome of the data class, storage class, management class, and storage group ACS routines must have a corresponding definition in the SCDS. For example, if you have five unique storage classes that can be determined by the storage class ACS routine, then you must have five corresponding storage class definitions in the SCDS.
- All classes and storage groups referenced in all ACS routines exist (issues a warning only).
- The storage group ACS routine exists.
- At least one VIO, pool, object type storage or tape group exists.
- All pool and dummy storage groups have at least one volume.
- At least one storage class exists.
- All pool and VIO storage groups are possible outcomes of the storage group ACS routine (issues a warning only).
- No dummy storage group is set by storage group ACS.
- All libraries associated with storage groups must exist in the configuration.
- A configuration can contain only one object backup storage group per system.
- Every optical library must have at least one drive associated with it.
- The DATABASE 2 (DB2) qualifier specified for each object or object backup storage group must be unique.
- Every object backup storage group must have one to eight real libraries or one to eight pseudo libraries, but not both.
- Every object storage group must have from one to eight real libraries or one to eight pseudo libraries, but not both.
- An object or object backup storage group cannot reference a tape library.
- A tape storage group cannot reference an optical library.
- A tape storage group must have at least one tape library.
- If a default entry data class is specified, the data class must exist in the configuration.
- All cache sets specified in storage classes must be defined in the base configuration

Translating and Validating in a Multiprocessor Environment

When running in an SMS multiprocessor complex with mixed levels of MVS/DFP (3.1 and higher), you must do the following on the system running the highest level of DFSMS/MVS:

- | Define new SMS constructs and ACS routines
- | Modify existing SMS constructs and ACS routines
- Translate your ACS routines
- Validate your SCDS

You can then activate and share the validated SCDS among systems with mixed levels of DFSMS/MVS. If you do not translate your ACS routines and validate your SCDS on the highest level of DFSMS/MVS, the translation and validation might fail. This failure occurs because some read-only variables are only known to higher levels of DFSMS/MVS. It can also occur due to changes in validation rules between releases.

Compatibility PTFs are needed for lower levels of DFP to share CDSs with DFSMS/MVS 1.3 and above. However, sharing among systems is only possible if you run your DFSMS/MVS 1.3 or above system using compatibility mode.

Note that while compatibility PTFs allow you to share CDSs among mixed levels of DFSMS/MVS and switch back to a prior release, once the CDSs have been formatted by the higher-level system, the CDS control blocks reflect the new rather than the downlevel lengths.

When sharing the same SMS CDSs among mixed releases, or when switching back to a lower-level release using the same SMS CDSs which were formatted by the higher-level system, all compatibility PTFs *must* be applied to avoid corrupting the CDSs.

Testing ACS Routines

After completing your ACS routines, you can write and execute test cases using the ISMF Automatic Class Selection application. After testing the individual routines of a new or modified configuration, you can activate it with greater confidence. ACS installation exits are not invoked during ACS routine testing.

Note that during ACS testing, only those parameters that are passed to the ACS routines at ACS time are tested. For some data processing, what is specified in the JCL is not necessarily what is passed to the ACS routines. A good example of this is how the RETPD and EXPDT parameters work: if EXPDT is specified in the JCL, during normal processing the RETPD read-only variable is set based on the EXPDT. If your ACS routine logic bases decisions on RETPD variable, and you specify EXPDT as a test parameter, you might get unexpected test results. ACS testing is not designed to simulate all of the possible processing that might occur before ACS processing. You must take care to know what is passed to the ACS routines at ACS time to ensure that you are testing what you expect to occur on your system.

ACS Test Usage

Since a DSNTYPE of EXT cannot be specified in JCL, ACS test for an data set allocated in extended format has some considerations.

To test data class only, you should not specify DSNTYPE = EXT for extended format data sets since the value of DSNTYPE in the Data Class ACS routine is never set to EXT during actual ACS processing.

To test more than just the Data Class for extended format data sets, you can choose to test the Data Class ACS routine with a separate test case that has no DSNTYPE specified. To test other constructs, such as management class, storage class and storage group, you should use a test case with DSNTYPE set to EXT. Note that, unlike actual ACS processing, ACS read-only variables are not defaulted from Data Class when executing the ACS test case.

Creating ACS Test Cases

If you wish to test ACS routines, you need to allocate a partitioned data set for the ACS test cases. The partitioned data set must have a logical record length of 80 or greater and a record format of F or FB. After allocating the data set, go to the ISMF Primary Option Menu and select option 7, Automatic Class Selection Application. To specify test cases, select option 4, Test, from the Automatic Class Selection Application Selection panel. When you press ENTER, you get the ACS Test Selection panel shown in Figure 62.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                              ACS TEST SELECTION
Command ===>


Select one of the following Options:

  1  1. DEFINE      - Define an ACS Test Case
     2. ALTER       - Alter an ACS Test Case
     3. TEST        - Test ACS Routines


If DEFINE or ALTER Option is Chosen, Specify:

  ACS Test Library . . SCDS1.ACSTEST.LIB
  ACS Test Member  . . TEST1






Use ENTER to Perform Selection;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 62. ACS Test Case Selection panel

The ACS Test Library field identifies the data set that contains the ACS test case. The ACS Test Member field identifies which member of the ACS test library contains the desired test case. You can specify one test case per member. After specifying a test library and member, select option 1 and press ENTER to specify test criteria. You see page one of the ACS Test Case Define panel, shown in Figure 63. This test case checks to see that the database storage group is a possible outcome of the storage group ACS routine.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                            ACS TEST CASE DEFINE                Page 1 of 3
Command ===>

ACS Test Library  : SCDS1.ACSTEST.LIB
ACS Test Member . : TEST1

To DEFINE ACS Test Case, Specify:
  Description ==> _
  Expected Result _
  DSN (DSN/Collection Name) . .
  MEMN (Object Name)  . . . . .
  Sysname . . .          Xmode . . . .             Def_dataclas  . .
  Sysplex . . .          ACSenvir  . .             Def_mgmtclas  . .
  DD  . . . . .          Dataclas  . .             Def_storclas  . .
  Dsorg . . . .          Mgmtclas  . .             Dsntype . . . . .
  Recorg  . . .          Storclas  . .               If Ext  . . . .
  Dstype  . . .          Storgrp . . .
  Dsowner . . .          Size  . . . .
  Expdt . . . .          Maxsize . . .
  Retpd . . . .
Use ENTER to Perform Verification; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 63. Creating ACS Test Cases, Page 1 of 4

Figure 64 shows page 2 of the ACS Test Case Define panel:

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                            ACS TEST CASE DEFINE                Page 2 of 3
Command ===>


ACS Test Library  : SCDS1.ACSTEST.LIB
ACS Test Member . : TEST1

To DEFINE ACS Test Case, Specify:
  Job  . . . . .          Pgm . . . . .             Vol . . . . .
  Group  . . . .          Applic  . . .             Unit  . . . .
  User . . . . .          Nvol  . . . .             Msvgp . . . .
  Acct_job  ===>
            ===>
            ===>
  Acct_step ===>
            ===>
            ===>
  Label  . . . .
  Filenum  . . .
  Libname  . . .

Use ENTER to Perform Verification; Use UP/DOWN Command to View other Panels;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 64. Creating ACS Test Cases, Page 2 of 4

Figure 65 on page 154 shows page 3 of the ACS Test Case Define Panel:

```
  Panel  Utilities  Scroll  Help
-----------------------------------------------------------------------------
                            ACS TEST CASE DEFINE                Page 3 of 4
Command ===>

ACS Test Library  : USER2.TESTLIB
ACS Test Member . : TEST11

To DEFINE ACS Test Case, Specify:

  Dest . . . . . ____________________________________________
  Pool . . . . . ________
  Policy . . . . ________
  Parm     ====> __________________________________________________
           ====> __________________________________________________
           ====> ____________________________________________





Use ENTER to Perform Verification; Use UP/DOWN Command to View other Panels;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 65. Creating ACS Test Cases, Page 3 of 4

On this panel, you can specify values corresponding to the MSxxxx parameters which you expect to be returned from your tape management system through the pre-ACS routine. See “Tape Management System Support” on page 164 for a description of these parameters, and use the documentation provided by your tape management vendor for more specifics.

Figure 66 shows page 4 of the ACS Test Case Define Panel:
On these four panels, you can specify test values that correspond to the ACS

```
  Panel  Utilities  Scroll  Help
-------------------------------------------------------------------------------
                   ACS TEST CASE VOLUME SERIAL SPECIFICATION       Page 4 of 4
Command ===>


ACS Test Library  : SCDS1.ACSTEST.LIB
ACS Test Member . : TEST1

To Complete ACS Test Case DEFINE, Specify:
  Volser ===>         ===>         ===>         ===>         ===>
  Volser ===>         ===>         ===>         ===>         ===>
  Volser ===>         ===>         ===>         ===>         ===>
  Volser ===>         ===>         ===>         ===>         ===>
  Volser ===>         ===>         ===>         ===>         ===>
  Volser ===>         ===>         ===>         ===>         ===>
  Volser ===>         ===>         ===>         ===>         ===>
  Volser ===>         ===>         ===>         ===>         ===>
  Volser ===>         ===>         ===>         ===>         ===>
  Volser ===>         ===>         ===>         ===>         ===>
  Volser ===>         ===>         ===>         ===>         ===>
  Volser ===>         ===>         ===>

Use ENTER to Perform Verification; Use UP Command to View previous Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 66. Creating ACS Test Cases, Page 4 of 4

variables listed in “Chapter 15. ACS Language Reference” on page 257. Use the

END command to save your test values and return to the ACS Test Selection panel. In the Test Selection panel, you can add more test case members to the same test library, or you can build another test library.

Running ACS Test Cases

To run the test cases in a test library, specify the library name and select option 3 in the ACS Test Selection panel (Figure 62 on page 152). You receive the Test ACS Routines panel shown in Figure 67.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                              TEST ACS ROUTINES
Command ===>


To Perform ACS Testing, Specify:

  CDS Name . . . . . . 'SMS.SCDS1.SCDS'
                                (1 to 44 Character Data Set Name or 'Active')
  ACS Test Library . . 'SCDS1.ACSTEST.LIB'
  ACS Test Member  . . TEST1     (fully or partially specified or * for all
                                members)
  Listing Data Set . . 'TEST1.LISTING'
                                (1 to 44 Character Data Set Name or Blank)

Select which ACS Routines to Test:

  DC ===> Y  (Y/N)  SC ===> Y  (Y/N)  MC ===> Y  (Y/N)  SG ===> Y  (Y/N)




Use ENTER to Perform Verification and Testing;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 67. Testing an ACS Routine

On this panel, you indicate which routines you want to test. Specifying an asterisk in the ACS Test Member field is convenient for running all test cases in the partitioned data set library. You can create a listing of the results of the testing by specifying a data set name in the Listing Data Set field. Leaving this field blank prevents the creation of a listing.

You can now test the routines by pressing ENTER. If you specified a listing data set, the results of the testing are displayed in the PDF Browse panel shown in Figure 68 on page 156.

```
 BROWSE -- TEST1.LISTING ---------------------------- LINE 00000000 COL 001 080
 COMMAND ===>                                                  SCROLL ===> PAGE
********************************* TOP OF DATA  *********************************
                               ACS TESTING RESULTS

CDS NAME         : 'SMS.SCDS1.SCDS'

ACS ROUTINE TYPES: SG

ACS TEST LIBRARY : SCDS1.ACSTEST.LIB

  ACS TEST
  MEMBER       EXIT CODE   RESULTS
  ---------    ----------  ------------------------------------
  TEST1                 0  SG = DATABASE


ACS TESTING RC:  00
******************************** BOTTOM OF DATA ********************************
```

Figure 68. Results of Testing an ACS Routine

After examining the results, issue the END command and specify on the ACS Output Listing Disposition whether or not to keep the output listing.

ACS Routines Invoked for Copying and Importing Data Sets

Table 7 shows which ACS routines are invoked when performing initial allocation, importing, copying, restoring and recalling data sets.

Table 7. Allocation, IMPORT, and COPY Conditions

Type of Processing	Data Class ACS	Storage Class ACS	Management Class ACS	Storage Group ACS
Initial Allocation	Yes	Yes	SC	SC
IMPORT (Access Method Services)	No	Yes	SC	SC
COPY (DFSMSdss)	No	Yes	SC	SC
COPY BYPASSACS (DFSMSdss)	No	No	No	SC

Yes—ACS routine is invoked

No—ACS routine is not invoked [1]

SC—ACS routine is invoked only if storage class is assigned

Notes:

1. The ACS routine is not invoked for the data set that is being copied or imported as their attributes are already defined. The ACS routine might be invoked for other new data sets allocated to the job.

Initial Allocation

Because the data set is new in an initial allocation, the data class ACS routine is invoked to determine the characteristics of the data set.

The storage class ACS routine is then invoked to determine whether the data set is system-managed.

The management class ACS and storage group ACS routines are invoked only if the data set is system-managed and a storage class was assigned.

Each time a dynamic allocation is issued by application programs, such as DFSMShsm RECALL, SMS follows this initial allocation path (except for the data class ACS routine which is not invoked).

IMPORT (Access Method Services)

The IMPORT (IDCAMS) command has three conditions:
- The VSAM data set already exists
- The VSAM data set is preallocated (allocated but empty)
- The target data set is created by the IDCAMS IMPORT job

The data is copied from a backup copy or another VSAM data set, and is imported to the target data set. If a storage class is assigned, the data is then copied into a system-managed target data set. Otherwise it is copied into a non-system-managed target data set.

The data class ACS routine is not invoked, because either the data set already exists, or the characteristics of the data set are derived from the IDCAMS IMPORT job.

The storage class ACS routine is invoked to determine whether the data set is system-managed. The imported data might come from a non-system-managed source data set that was then copied into a system-managed target data set.

The management class ACS and storage group ACS routines are invoked only if the data set is system-managed and a storage class was assigned.

COPY (DFSMSdss)

The COPY command has three conditions:
- The target data set already exists
- The target data is preallocated (allocated but empty)
- The target data is created by the DFSMSdss COPY job

The data is either copied from a backup copy of the data set, or from another data set to the target data set.

The data class ACS routine is not invoked, because either the data set already exists, or the characteristics of the data set are derived from the DFSMSdss COPY job.

The storage class ACS routine is invoked to determine whether the data set is system-managed. The copied data might come from a non-system-managed source data set that was then copied to a system-managed target data set.

The management class ACS and storage group ACS routines are invoked only if the data set is system-managed.

See *DFSMS/MVS DFSMSdss Storage Administration Guide* for more information on the DFSMSdss COPY command.

COPY BYPASSACS (DFSMSdss)

The COPY BYPASSACS command has three conditions:
- The target data set already exists

- The target data is preallocated (allocated but empty)
- The target data is created by the DFSMSdss COPY job

The data is copied from either a backup copy or another data set to the target data set. If a storage class is assigned, the data is then copied into a system-managed target data set. Otherwise it is copied into a non-system-managed target data set.

The data class, storage class, and management class ACS routines are not invoked because the BYPASSACS keyword is coded in the DFSMSdss COPY job.

The storage class ACS routine is invoked only if it is specified in the DFSMSdss COPY job.

The storage group ACS routine is invoked only if SC is specified in the DFSMSdss COPY job, or if the data set is system-managed.

Note: You need RACF authorization to use the BYPASSACS keyword. The storage or security administrator must define BYPASSACS to RACF as a facility and then tell RACF who is authorized to use it.

ACS Routines Invoked for Restoring, Recalling, Recovering, and Converting Data Sets

Table 8 shows which ACS routines are invoked for restoring, recalling, recovering, and converting data sets.

Table 8. RESTORE, RECALL, RECOVER, CONVERTV, and FORCENONSMS Conditions

Type of Processing	Data Class ACS	Storage Class ACS	Management Class ACS	Storage Group ACS
DFSMSdss RESTORE	No	Yes	SC	SC
DFSMSdss RESTORE BYPASSACS	No	No	No	SC
DFSMSdss CONVERTV	No	Yes	SC	No
DFSMShsm RECALL/ RECOVER	No	Yes	SC	SC
DFSMShsm FORCENONSMS	No	No	No	No

Yes—ACS routine is invoked

No—ACS routine is not invoked [1]

SC—ACS routine is invoked only if storage class is assigned

Notes:

1. The ACS routine is not invoked for the data sets as their attributes are already defined. The ACS routine might be invoked for other new data sets allocated to the job.

DFSMSdss RESTORE

When using the DFSMSdss RESTORE command, the data set either exists or is deleted. The data set is recovered by restoring data from a backup copy to the target data set.

The data class ACS routine is not invoked, because either the data set already exists, or the characteristics of the data set are derived from the backup data set or DFSMSdss RESTORE job.

The storage class ACS routine is invoked to determine whether the data set is system-managed.

The management class ACS and storage group ACS routines are invoked only if the data set is system-managed and a storage class is assigned.

DFSMSdss RESTORE BYPASSACS

When using the DFSMSdss RESTORE BYPASSACS command, the data set either exists or has been deleted. The data set is recovered by restoring data from a backup copy to the target data set.

The data class, storage class, and management class ACS routines are not invoked because the BYPASSACS keyword is coded in the DFSMSdss RESTORE job.

The storage group ACS routine is only invoked if:
- Storage class is specified in the DFSMSdss RESTORE job.
- The data set is system-managed.

Note: You need RACF authorization to use the BYPASSACS keyword. Define BYPASSACS to RACF as a facility and then tell RACF who is authorized to use it.

DFSMSdss CONVERTV

When using the DFSMSdss CONVERTV command, the data class ACS routine is not invoked because the data sets on the volume already exist.

The storage class ACS routine is invoked because the data sets and volume are to become system-managed.

The management class ACS routine is invoked, if the storage class is assigned, to determine the appropriate management class names assigned to the data sets.

The storage group ACS routine is not invoked because all the data sets on the system-managed volume are assigned to the storage group to which the volume belongs.

DFSMShsm RECALL/RECOVER

When using the RECALL/RECOVER command, if you want to RECALL a data set, then the data set already exists. If you want to RECOVER a data set, then either the data set exists or has been deleted. The data set is recovered by restoring data from a DFSMShsm backup copy to the target data set.

The data class ACS routine is not invoked, because either the data set already exists, or the characteristics of the data set are derived from the backup copy.

The storage class ACS routine is invoked to determine whether the data set is system-managed.

The restored data might come from a non-system-managed backup copy, and is allocated as system-managed or non-system-managed as determined by the ACS routines.

The management class ACS and storage group ACS routines are invoked only if the data set is system-managed.

DFSMShsm FORCENONSMS

When using the FORCENONSMS keyword, if you want to RECALL a data set, then the data set already exists. If you want to RECOVER a data set, then either the data set exists or is deleted. The data set is recovered by restoring data from a DFSMShsm backup copy to the target data set.

Because the data set is not supposed to be system-managed, no ACS routines are invoked.

ACS Routine Environments

Depending on the environment, SMS invokes some or all of the ACS routines in the following order:

1. Data class
2. Storage class
3. Management class
4. Storage group

JCL DD Statement (Batch), and Dynamic Allocation

For the &ACSENVIR='ALLOC' environment, the data class ACS routine and then the storage class ACS routine are executed. If the storage class is not null, the management class ACS routine and then the storage group ACS routine are executed.

Volume Reference

You can code VOL=REF on a DD statement to refer to a DD statement in the same or an earlier step to allocate the data set on the same volume as the earlier data set. You can also code VOL=REF to an existing data set (VOL=REF=A.B.C,...) where A.B.C is a cataloged data set. If VOL=REF is coded, SMS invokes ACS routines as follows:

- SMS invokes the data class ACS routine.
- Storage class is copied from referenced data set if the referenced data set has a storage class assigned to it.

  Notes:

  1. This is not true for non-SMS data sets, since they do not have storage classes to copy. In these cases, the routine is invoked and can either allow the non-SMS allocation or fail it. If it does anything else, SMS fails it.
  2. This is also not true for SMS-managed tape data sets, where the storage class is not saved. In these cases, VOL=REF processing must use the following read-only variables: &LIBNAME, &ANYVOL, &ALLVOL.
  3. Because of the tendency to copy JCL, the VOL=REF to a non-SMS-managed data set might be used when the data set is allocated as old (DISP=OLD or

DISP=SHR). Since the VOL=REF is used in locating an existing data set, allowing the data set to be allocated as SMS-managed when it is new could cause problems at this time.

4. If you use VOL=REF processing to refer to a temporary data set, you might get different results in storage group assignments than expected. This is because temporary data sets are assigned a storage group by the system, based on a list of eligible storage groups, such as: VIO, PRIME, STANDARD, etc. Data sets that use VOL=REF are assigned a storage group based on this list of eligible storage groups, not on the name of the storage group used to successfully allocate the first data set being referenced. This might result in the data sets being allocated in different storage groups.

- Management class ACS routine is called if storage class is not null.
- The storage group from the referenced data set is passed as input to the storage group ACS routine. For data sets on SMS-managed tape volumes, the ACS routine must assign the same storage group to the referencing data set. For other SMS-managed data sets, any pool or VIO storage group can be assigned to the referencing data set.

Note: Starting with DFSMS/MVS 1.3, ACS routines receive control for new data sets allocated with VOL=REF specified and that point to a non-SMS-managed data set. Prior to DFSMS/MVS 1.3, ACS routines did *not* receive such control.

When VOL=REF is used, the &ALLVOL and &ANYVOL ACS read-only variables are set to 'REF=SD,' 'REF=ST' or 'REF=NS' as appropriate. Additionally, if the reference is to a data set on an SMS-managed volume, the storage group of the referenced data set is provided in the &STORGRP read-write variable, if it is available. (For some references to data sets on SMS-managed tape, it might not be, in which case the ACS routine should use the value of the &LIBNAME read-only variable instead.) If the referenced data set is new, it can still have multiple candidate storage groups since it hasn't been allocated yet. In that case only the *first* candidate storage group is passed in.

The &ALLVOL and &ANYVOL ACS read-only variables contain the following values when you use VOL=REF:

- 'REF=SD' (the volume reference is to an SMS-managed DASD or VIO data set)
- 'REF=ST' (the volume reference is to an SMS-manged tape data set)

  Figure 69 on page 162 illustrates these values:

```
PROC STORGRP

SELECT(&ANYVOL)
  WHEN('REF=SD')
    IF &DSTYPE = 'TEMP' & &&DSORG ^= 'VS' THEN
      SET &STORGRP = 'VIOSG','MAIN3380','MAIN3390','SPIL3380','SPIL3390'
    ELSE
      SET &STORGRP = 'MAIN3380','MAIN3390','SPIL3380','SPIL3390'
  WHEN('REF=ST')
    SET &STORGRP = &STORGRP
  OTHERWISE
  .
  .
  .
END
END
```

Figure 69. Example of REF=ST Values when Using VOL=REF

- 'REF=NS' (the volume references is to a non-SMS-managed data set)

  Figure 70 illustrates these values:

  As of the date shown in Figure 70, you might change the ACS routine to fail the

```
PROC 0 STORCLAS
FILTLIST AUTH_USER INCLUDE('SYSPROG1','SYSPROG2','STGADMIN','SYSADMIN')

IF &ANYVOL = 'REF=NS' & &HLQ ^= 'SYS1' THEN
  IF &USER ^= &AUTH_USER THEN
    DO
      WRITE 'INVALID USE OF VOL=REF TO A NON-SMS-MANAGED DATA SET'
      WRITE 'AS OF 12/31/94 ALLOCATION WILL BE FAILED FOR ' &DSN
    END
  .
  .
  .
END
```

Figure 70. Example of REF=NS Values when Using VOL=REF

  uses of VOL=REF that are not valid, as shown in Figure 71.

```
PROC 1 STORCLAS
FILTLIST AUTH_USER INCLUDE('SYSPROG1','SYSPROG2','STGADMIN','SYSADMIN')

IF &ANYVOL = 'REF=NS' & &HLQ ^= 'SYS1' THEN
  IF &USER ^= &AUTH_USER THEN
    DO
      WRITE 'INVALID USE OF VOL=REF TO A NON-SMS-MANAGED DATA SET'
      WRITE 'DATA SET ' &DSN ' MUST BE SMS-MANAGED'
      EXIT CODE(4)
    END
  .
  .
  .
END
```

Figure 71. Example of Failing VOL=REF Values That Are Not Valid

Data Set Stacking

Data set stacking is the function used to place several data sets on the same tape volume or set of tape volumes. It increases efficiency when using tape media and reduces the overall number of tape volumes needed by allocation. It also allows an installation to group related data offsite. The data set sequence number subparameter on the JCL LABEL parameter is used in conjunction with VOL=REF or VOL=SER to accomplish this function.

Under certain conditions, SMS invokes the ACS routines more than once. This section describes those conditions as well as the values of the read-only variables for the initial invocation and any subsequent invocations of the ACS routines. Since data set stacking might cause a second or third invocation of the ACS routines, you might want to take special care when using WRITE statements to avoid duplicates in the job log.

Using VOL=SER Within a Job Step

When data set stacking is specified using VOL=SER within a job step, the system ensures that all the data sets making up the data set collection (a group of data sets intended to be allocated on the same tape volume or set of tape volumes as a result of data set stacking) are directed to the same device category. If the ACS routines initially directed the stacked allocations to different device categories, the system detects this and reinvokes the ACS routines, passing additional information to those routines. The ACS routines can then do one of the following:

- Correct the problem and route the allocations to consistent device categories
- Fail the stacked allocation (if the ACS routine exits with a non-zero return code)
- Fail to correct the inconsistency, in which case SMS fails the allocation

Note: The system cannot detect data set stacking if VOL=SER is used to stack data sets across jobs or job steps. In these instances, you can change your JCL to specify VOL=REF instead of VOL=SER.

For more recommendations on when to use VOL=REF versus VOL=SER for data set stacking, see *OS/390 MVS JCL User's Guide*.

The system reinvokes the ACS routines only when all of the following conditions are true:

- The request is part of a data set collection based on a data-set-sequence-number greater than one specified on the LABEL parameter, and a VOL=SER, where at least one of the volume serial numbers matches one of the volume serial numbers for a previous request in the same step.
- The request is currently directed to a different device category than the other requests in the data set collection.
- The request is DISP=NEW (or DISP=MOD treated as NEW).

Using VOL=REF

When data set stacking is requested with the VOL=REF parameter, the ACS routines are passed information that indicates that volume reference is used. Therefore, the ACS routines can direct the requests within a data collection to the same device category.

Possible Values for the &UNIT Read-Only Variable

Additional information is now passed to the ACS routines in the &UNIT ACS read-only variable, so that the ACS routines know when data set stacking or unit affinity is used. In a tape environment, unit affinity is a JCL keyword (UNIT=AFF) used to minimize the number of tape drives used in a job step.

Table 9 shows the values to which the &UNIT read-only variable can be set. The first value is applicable only when UNIT=AFF is used. The others are applicable only when data set stacking is being done, whether or not the UNIT=AFF keyword is present.

Table 9. Values for &UNIT ACS Read-Only Variable

&UNIT Value	ACS Invocation	Data Set Stacking Indication	Device Category of Data Set on Which to Stack
AFF=	First	Unknown	Not applicable
STK=SMSD	Second	Yes and different device categories	System-managed DASD
STK=NSMS	Second	Yes and different device categories	Non-system-managed DASD or Non-system-managed Tape
STK=SMSD or STK=NSMS	Third	Yes and different device categories	Non-system-managed DASD or Non-system-managed Tape

Note: ACS routines can be invoked three times in a JES3 environment.

Figure 72 shows an example of a storage class ACS routine for read-only variables:

```
PROC  &STORCLAS

SELECT(&UNIT)
  WHEN('STK=SMSD')
    SET &STORCLAS = 'POOLSC'
  WHEN('STK=NSMS')
    SET &STORCLAS = '
  OTHERWISE
```

Figure 72. Example of a Storage Class ACS Routine for Read-Only Variables

The storage group ACS routine could then do something like what is shown in Figure 73:

```
PROC  &STORGRP

SELECT(&UNIT)
  WHEN('STK=SMSD')
    SET &STORGRP = 'S1P03'
  WHEN('STK=NSMS')
    SET &STORGRP = 'POOLSG'
  OTHERWISE
```

Figure 73. Example of a Storage Group ACS Routine for Read-Only Variables

| ## Tape Management System Support

| To support your tape management system's need to coordinate complex vaulting
| requirements with data set allocation in system-managed environments, you can
| use a pre-ACS routine exit to set values for read-only variables which SMS then
| uses as input to ACS routines. This support is useful when the choice of a storage
| group needs to be influenced by the data set's vaulting requirements. For example,

you might use this support when excluding data from redirection under tape mount management or directing data to a specific system-managed library[1].

The four read-only variables used to support tape management system—driven tape allocations are:

&MSDEST
: used to specify a destination, in data set name format. This format lets you specify a sequence of destinations to be identified, where each qualifier is a specific destination. For example, a data set vaulted first at location OUTD and then sent to OLTS could have an MSDEST of `'OUTD.OLTS'`. The actual values depend on the support provided by your tape management system.

&MSPARM
: used to specify any additional information. This is a variable length field.

&MSPOLICY
: used to identify a management policy associated with data in a Virtual Tape Server (VTS).

&MSPOOL
: used to specify a tape pool name associated with the data set being allocated. In a system-managed tape environment with scratch pool support, this variable might be used to specify a default storage group, where the tape storage group is equivalent to the tape pool specified in the variable.

Access Method Services

For the ALLOCATE and DEFINE commands (&ACSENVIR='ALLOC'), the data class ACS routine and then the storage class ACS routine are executed. If the storage class is not null, the management class ACS routine and then the storage group ACS routine are executed.

For the IMPORT command, the storage class ACS routine is executed first. If the storage class is not null, the management class ACS routine and then the storage group ACS routine are executed.

Redrive MGMTCLAS ACS routine on data set rename invokes the management class ACS routine when an SMS managed cluster, GDS or non-VSAM data set is renamed. Catalog management invokes MGMTCLAS ACS routines during RENAME processing. You can reassign a different management class based on the following data set attributes:

- New data set name
- Data set type
- Data set organization
- Expiration date in the catalog
- Old management class
- Data class
- Storage class
- Record organization
- User information

1. Import/Export support is available with APAR OW36342 or OW36343. If you are planning on using the Pre-ACS routine exit routine, APAR OW36351 or OW36352 should be installed.

- Group information

ACS routines support RENAME. The following read-only variables are set for the management class ACS routine:

- &ACSENVIR:RENAME
- &DSN of the new data set name
- &DSORG
- &DSTYPE
- &EXPDT
- &STORCLAS
- &DATACLAS
- &MGMTCLAS
- &GROUP of the job
- &USER of the job
- &HLQ
- &LLQ

DFSMShsm

The management policy during a data set recall or recover is determined by the value set for &ACSENVIR in the management class ACS routine. The storage class ACS routine is then invoked to apply performance and availability criteria. The data class routine is not invoked. Additionally, during DFSMShsm RECALL processing, the storage group routine gets a different value for the &ACSENVIR field, depending on the data mover that is used. If DFSMSdss is the data mover, RECOVER is passed as the environment; if DFSMShsm is the data mover, RECALL is passed as the environment. Because of this, do not use the storage group ACS routine to test the value in &ACSENVIR for recall or recover, as this could yield inconsistent results.

DFSMSdss

For the COPY command (&ACSENVIR='ALLOC') and the RESTORE command (&ACSENVIR='RECOVER'), the storage class ACS routine is executed first. If the storage class is not null, the management class ACS routine and then the storage group ACS routine are executed.

For CONVERTV SMS TEST and CONVERTV SMS, (&ACSENVIR='CONVERT'), the storage class ACS routine is executed first. If the storage class ACS routine determines that the storage class is not null, the management class ACS routine is executed.

ISMF

When you are testing ACS routines, the data class ACS routine and then the storage class ACS routine are executed. If the storage class ACS routine determines that the storage class is not null, the management class ACS routine and then the storage group ACS routine are executed. You can also execute each ACS routine separately from the others when performing tests.

OAM

For the OSREQ CHANGE command (&ACSENVIR='CHANGE'):

- If storage class, or both storage class and management class, are specified, both the storage class and management class ACS routines are executed, in that order.
- If only management class is specified, only the management class ACS routine is executed, with the old storage class used as input.

For the OSREQ STORE command (&ACSENVIR='STORE'):

- If the object is the first object in a collection, the storage class, management class and storage group ACS routines are executed, in that order. These routines determine the defaults for the object collection.
- Whenever storage class or management class is specified, the storage class and management class ACS routines are executed for the object with the specified classes as input. Then the ACS routines derive the initial storage class and management class for the object rather than having the object use the default initial storage and management classes for the collection.

At class transition time, as defined by the management class associated with an object, (&ACSENVIR='CTRANS'), the storage class routine and then the management class routine are executed.

Processing of SMS Classes and Storage Groups

For each of the SMS classes, the processing is as follows:

1. If you have an ACS routine in your CDS to determine the SMS class, SMS executes the routine.
2. Next, SMS executes the corresponding *ACS installation exit*. An ACS installation exit is an assembler language program you can write to perform processing beyond the scope of the standard ACS routines. Such processing might involve:
   - Calling other programs
   - Writing SMF records
   - Writing generalized trace facility (GTF) trace records
   - Performing arithmetic calculations
   - Maintaining large tables of information for quick searches
   - Writing dumps
   - Invoking ACS routines only once

   Each exit can override the corresponding SMS class, whether explicitly specified or previously determined by an ACS routine. The exit can also invoke the corresponding ACS routine a second time, but this does not cause the installation exit to be re-invoked.

   See *DFSMS/MVS Installation Exits* for additional information about ACS installation exits.
3. Finally, if RACF is installed and the storage administrator has defined STORCLAS and MGMTCLAS as resources and permitted end users to specify them, the system verifies that the class is defined in the currently active configuration. The system then checks to verify that end users are allowed to use a selected management class or storage class.

For storage groups, SMS invokes only the storage group ACS routine. Storage group does not have a corresponding ACS installation exit.

Displaying ACS Object Information

To display information about the ACS objects stored in a control data set, select option 5, DISPLAY, from the Automatic Class Selection Application Selection panel and press ENTER. You receive the ACS Object Display panel shown in Figure 74.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                              ACS OBJECT DISPLAY
Command ===>


CDS Name  : SMS.SCDS1.SCDS

ACS Rtn   Source Data Set ACS      Member    Last Trans  Last Date  Last Time
Type      Routine Translated from  Name      Userid      Translated Translated
--------  -----------------------  --------  ----------  ---------- ----------
DATACLAS  STAGE2.SYSTEM1.ACSRTN.   DATACLAS  IBMUSER     1994/08/24  15:27
          SOURCE

MGMTCLAS  STAGE2.SYSTEM1.ACSRTN.   MGMTCLAS  IBMUSER     1993/11/04  16:53
          SOURCE

STORCLAS  STAGE2.SYSTEM1.ACSRTN.   STORCLAS  IBMUSER     1995/02/15  15:58
          SOURCE

STORGRP   STAGE2.SYSTEM1.ACSRTN.   STORGRP   IBMUSER     1993/11/04  16:53
          SOURCE


Use HELP Command for Help; Use END Command to Exit.
```

Figure 74. Displaying ACS Object Information

CDS NAME can be the name of the SCDS that you entered on the Automatic Class Selection Application Selection panel or 'ACTIVE'. It identifies the data set that contains the ACS objects that are to be displayed.

ACS RTN TYPE always lists the four types of ACS objects that can belong to an SCDS, even if they do not actually exist in the SCDS being displayed. If they do not exist, the corresponding display fields contain dashes.

SOURCE DATA SET ACS ROUTINE TRANSLATED FROM contains the name of the data set that has the source code responsible for creating the ACS object. The figure shows that the data set name is folded if it is more than 23 characters in length. If an ACS object of a particular type does not exist in the SCDS being displayed, then this field contains dashes.

MEMBER NAME displays the member name within the data set that contains the ACS source code used to create the corresponding ACS object. If an ACS object of the particular type does not exist in the SCDS being displayed, or if the SOURCE DATA SET ACS ROUTINE TRANSLATED FROM is not a PDS or PDSE, the field contains dashes.

LAST TRANS USERID displays the TSO user ID of the person who last translated the ACS routine. If an ACS object of the particular type does not exist, the field contains blanks.

LAST DATE TRANSLATED displays the date when the corresponding ACS object was created. If an ACS object of the particular type does not exist, the field contains blanks.

LAST TIME TRANSLATED displays the time when the corresponding ACS object was created. If an ACS object of the particular type does not exist, the field contains blanks.

Deleting an ACS Object from an SCDS

To delete an ACS object from an SCDS, select option 6, DELETE, from the Automatic Class Selection Application Selection panel and press ENTER. You receive the Delete ACS Object panel shown in Figure 75.
The SCDS Name field contains the name of the SCDS from which the ACS object

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                              DELETE ACS OBJECT
Command ===>


To Delete an ACS Object, Specify:

  SCDS Name  . . . . . 'SMS.SCDS1.SCDS'
                           (1 to 44 Character Data Set Name)

  ACS Routine Type . . SG  (DC=Data Class, MC=Management Class, SC=Storage
                           Class, SG=Storage Group)









Use ENTER to Perform ACS Deletion;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 75. Deleting ACS Objects

is to be deleted. It is a required field and is primed with the last referenced SCDS. The SCDS name cannot be ’ACTIVE’. The default is blank.

The ACS Routine Type field contains the type of ACS object. It is a required field and it is primed with the last used value. The default is blank.

When you press ENTER, you receive the Confirm Delete Request panel to certify that you want to delete the ACS object.

Chapter 10. Activating Storage Management Subsystem Configurations

You can manually activate an SMS configuration, or you can automatically activate it at IPL. The first part of this chapter explains how to perform the initial activation of an SMS configuration using a four-step manual approach. The second part of this chapter explains how to activate an SMS configuration automatically at future IPLs. The final part of this chapter explains how to change individual SMS parameters using the SETSMS operator command.

Note: When you activate an SMS configuration, you should ensure that all of the DASD volumes that belong to the configuration are initialized as SMS volumes. Otherwise, attempted allocations to an improperly initialized volume fail. However, initialization for tape volumes is no different for SMS-managed and non-SMS-managed volumes.

Manually Activating the First Storage Management Subsystem Configuration

IGDSSIIN is the subsystem initialization routine module for SMS. By omitting it from the SMS entry of IEFSSNxx for each system in the SMS complex, you can manually control the activation of an SMS configuration. Refer to "Chapter 2. Preparing for the Storage Management Subsystem" on page 9.

Step One: IPL Each System in the SMS Complex

After defining SMS as a subsystem to MVS, IPL each system in the SMS complex. The presence of the SMS entry in IEFSSNxx tells MVS to recognize SMS as a valid subsystem within each system. The absence of the IGDSSIIN module name in IEFSSNxx tells the system that you want to manually start SMS.

Step Two: Prepare One System

From one system in the SMS complex, issue the T SMS=*xx* command, where *xx* identifies IGDSMSxx as the SMS initialization control member of SYS1.PARMLIB. SMS uses the ACDS and COMMDS identified in IGDSMSxx to manage storage. Because the initial ACDS and COMMDS are empty, the system is activated with a *null configuration*. Keep in mind that a null configuration is only intended as a migration path.

Note: All systems in the SMS complex must be running in the same mode. All systems running releases of DFSMS/MVS or MVS/DFP prior to DFSMS/MVS 1.3 are considered to be running in compatibility (8-name) mode. See "Defining the Base Configuration" on page 27 for more information about compatibility mode.

Step Three: Activate the Configuration from One System

The configuration is only activated once for the SMS complex. It is not necessary to activate a configuration from every system in the SMS complex. After activating SMS with a null configuration, activate an SMS configuration contained in a valid

SCDS on the same system. You can use either the ISMF ACTIVATE command or the SETSMS operator command. Both procedures copy the contents of the SCDS to the ACDS specified in IGDSMSxx.

When an SMS control data set that supports only eight names is accessed for update on a system running in 32-name mode, you must convert the data set to a new, incompatible format in order to support 32 names. Confirm this conversion, using the operator console or ISMF. This conversion is permanent, so you should make copies of your control data sets before the system mode is converted. If any of the systems are pre-DFSMS/MVS 1.3, then all 1.3 or above systems must run in compatibility mode.

Activating with the ISMF ACTIVATE Command

On the Control Data Application Selection panel shown in Figure 7 on page 28, specify the name of an SCDS and issue the ACTIVATE command from the command line. A *Write to Programmer* message indicates if the activation is successful, provided you have WTPMSG in the TSO/E PROFILE.

Activating with the SETSMS Operator Command

From the operator console, issue the command:

```
SETSMS SCDS(dsname)
```

where `dsname` identifies the name of the SCDS to be activated. Using the SCDS defined earlier in this manual, the command would be:

```
SETSMS SCDS(SMS.SCDS1.SCDS)
```

Step Four: Activate SMS on the Other Systems

For the other systems in the SMS complex, use the T SMS=xx command to start SMS on those systems, using the SMS configuration identified in Step Three. In each system, IGDSMSxx specifies the name of the ACDS containing the SMS configuration. All of the IGDSMSxx members must point to the same ACDS. Because the ACDS is no longer empty, the systems use it (and the COMMDS) to manage storage.

Automatically Activating a Storage Management Subsystem Configuration

For each system in the SMS complex, update the SMS entry in IEFSSNxx to include the IGDSSIIN module name.

Make certain that each ID field identifies the IGDSMSxx member containing the name of the last used ACDS. All of the IGDSMSxx members must point to the same ACDS. At future IPLs, the SMS configuration contained in the ACDS is activated by all systems in the SMS complex. For example:

```
SMS,IGDSSIIN,'ID=02,PROMPT=DISPLAY'
```

indicates that the ACDS specified in IGDSMS02 contains the SMS configuration to be activated at future IPLs.

Changing Storage Management Subsystem Parameters

Once you have activated an SMS configuration, you can use the SETSMS operator command to change SMS parameters.

SETSMS Operator Command

The SETSMS operator command has the following options:

ACDS(*dsname***)**
: identifies the name of the data set containing the active configuration. If you omit *dsname*, the operator is prompted for a value.

ASID({*asid***|*})**
: limits tracing to a certain address space or permits it for all address spaces. Specify * if you want to trace all address spaces (providing SMS tracing is active). This is the default. You can enter up to 4 digits for the ASID keyword. If you leave off the leading zeroes, they are inserted.

BMFTIME({*nnn***|3600})**
: specifies the number of seconds between recording SMF record type 42, subtype 1 records for PDSE and HFS I/O (where the buffer hits are I/O requests which did not result in actual I/O being done). You can specify a value from 1 to 86399 (23 hours, 59 minutes, 59 seconds), and the default is 3600 (one hour).

CACHETIME({*nnn***|3600})**
: specifies the number of seconds between recording SMF records for device cache use. The **CACHETIME** parameter applies only to the volumes behind an IBM 3990 Storage Control with cache unit. You can specify a value from 1 to 86399 (23 hours, 59 minutes, 59 seconds), and the default is 3600 (one hour).

CF_TIME({*nnn***|3600})**
: indicates the number of seconds between recording SMF records for the coupling facility (both cache and lock). You can specify a value from 1 to 86399 (23 hours, 59 minutes, 59 seconds), and the default is 3600 (one hour). This keyword sets the interval time for the following SMF 42 subtypes:

  SUBTYPE 15
  : Coupling facility storage class average response time

  SUBTYPE 16
  : Coupling facility data set average response time

  SUBTYPE 17
  : Coupling facility lock structure activity

  SUBTYPE 18
  : Coupling facility cache partition summary

  SUBTYPE 19
  : SMSVSAM least recently used statistics summary

COMMDS(*dsname***)**
: identifies the name of the COMMDS. If you omit *dsname*, the operator is prompted for a value.

DEADLOCK_DETECTION({*iiii***|15,***kkkk***|4})**
: specifies the deadlock detection intervals used by the DFSMSdfp Storage

Management Locking Services. The first subparameter is the local deadlock detection cycle and specifies the interval in seconds for detecting deadlocks within a system. The second subparameter is the global deadlock detection cycle and specifies the interval for detecting deadlocks between systems. The value is specified as the number of local detection cycles that occur before global deadlock detection is initiated.

iiii one to four digit numeric value in the range 1-9999 that specifies the length in seconds of the local deadlock detection interval. The default is 15 seconds.

kkkk one to four digit numeric value in the range 1-9999 that specifies the number of local deadlock cycles that must complete before global deadlock detection occurs. The default is 4.

DESELECT({*event[,event][,...]***|ALL})**
deletes items from the list of events to be traced (if SMS tracing is active). DESELECT has no default. If you specify events that conflict in SELECT and DESELECT, the keyword that appears last has final authority.

The events that you can specify on SELECT and DESELECT are:

MODULE
SMS module entry or exit

DSTACK
SMS Data Set Stacking Service

SMSSJF
SMS/SJF interfaces

SMSSSI
SMS/SSI interfaces

ACSINT
ACS services interfaces

OPCMD
Operator commands

CONFC
Configuration change

CDSC Control data set changes

CONFS
SMS Configuration services

MSG SMS Message services

ERR SMS Error recovery and recording services

CONFR
Return data from an active configuration

CONFA
Activate a new configuration

ACSPRO
Perform ACS processing

IDAX SMS interpreter/dynamic allocation

DISP SMS disposition processing exit

CATG SMS catalog services

VOLREF
SMS VOLREF services

SCHEDP
SMS scheduling services (preallocate catalog orientation)

SCHEDS
SMS scheduling services (system select)

VTOCL
SMS VTOC/data set services (allocate existing data set)

VTOCD
SMS VTOC/data set services (delete existing data set)

VTOCR
SMS VTOC/data set services (rename existing data set)

VTOCC
SMS VTOC/data set services (allocate new data set)

VTOCA
SMS VTOC/data set services (add a volume to a data set)

RCD SMS Recording services

DCF SMS device control facility

DPN SMS device pool name select subsystem interface

TVR SMS tape volume record update facility

ALL All of the above options

DINTERVAL({*nnn*|150})
specifies the number of seconds SMS allows to elapse before it reads device statistics (SMS uses device statistics to manage hardware usage and maximize efficiency). The **DINTERVAL** parameter applies only to the volumes behind an IBM 3990 Storage Control with cache unit. You can specify a value from 1 to 999 (16 minutes, 39 seconds), and the default is 150.

INTERVAL({*nnn*|15})
specifies the *synchronization time interval* of the system, which is the number of seconds between system checks of the COMMDS for news about SMS configuration changes from other systems in the SMS complex. You can specify a value from 1 to 999 (16 minutes, 39 seconds), and the default is 15.

JOBNAME({*jobname*|*})
limits tracing to a certain job or permits tracing on all jobs. The default is to trace all jobs, *****. This keyword supports objects or tape libraries.

RLSINIT({NO|YES})
specify YES if you want the SMSVSAM address space started as part of system initialization or the V SMS,SMSVSAM,ACTIVE command. This value applies only to the system accessed by the parmlib member and is acted upon when SMSVSAM is next started. The default is NO.

RLS_MAX_POOL_SIZE({*nnnn*|100})
specifies the maximum size in megabytes of the SMSVSAM local buffer pool. SMSVSAM attempts to not exceed the buffer pool size you specify,

although more storage might be temporarily used. Because SMSVSAM manages buffer pool space dynamically, this value does not set a static size for the buffer pool.

Use SMF 42, subtype 19 records to help you determine the maximum size of the SMSVSAM local buffer pool.

You can specify a two to four-digit numeric value, with 10 as the minimum value. If you specify a value less than 10, the field is set to 10. If you specify a value greater than 1500, SMSVSAM assumes there is no maximum limit. We recommend that you limit the size of the local buffer pool.

The default is **100MB**.

SELECT({*event[,event][,...***]|ALL})**
adds items to the list of events to be traced (if SMS tracing is active). The default is **ALL**. See **DESELECT** for a list of valid events.

SIZE(*nnn***{K|M})**
specifies the size of the SMS trace table in bytes. When the unit is **K**, the value can range from 0K to 255000K, and it is rounded up to the nearest 4K unit. When the unit is **M**, the value can range from 0M to 255M. If you specify a value of 0, no tracing is performed. The default is 128K.

SMF_TIME({YES|NO})
YES indicates that the following SMF type 42 records are created at the SMF interval time, and that all of the indicated records are synchronized with SMF and RMF data intervals:

SUBTYPE 1
Buffer management statistics

SUBTYPE 2
Cache control unit statistics (IBM 3990 Storage Control Model 3)

SUBTYPE 15
Coupling facility storage class average response time

SUBTYPE 16
Coupling facility data set average response time

SUBTYPE 17
Coupling facility lock structure activity

SUBTYPE 18
Coupling facility cache partition summary

SUBTYPE 19
SMSVSAM least recently used statistics summary

DFSMS creates the specified SMF record at the end of the interval period and SMF sends the event notification signal. If YES is specified, this subparameter overrides the following subparameters: BMFTIME, CACHETIME, CF_TIME. YES is the default.

See *OS/390 MVS System Management Facilities (SMF)* for more information on SMF.

TRACE({OFF|ON})
indicates whether the SMS trace facility is activated. The default is **ON**. The SMS trace facility records trace records in the SMS address space. Refer to *DFSMS/MVS DFSMSdfp Diagnosis Reference* for additional information.

TRACEEXIT(*user_trace_exit***)**
: specifies the name of your trace exit routine. You can specify an alphanumeric value of 1 to 8 characters, starting with an alphabetic character. Your value must represent a valid module name. The default value is blank. This interface is usually used only for diagnostic purposes under the guidance of the service organization and therefore is not documented here.

TYPE({ALL|ERROR})
: specifies whether SMS traces all events or only errors. The default is **ERROR**.

If you do not specify a keyword value, SMS uses the current value (specified in IGDSMSxx or by a previous operator command).

Considerations when Changing Storage Management Subsystem Configurations

When activating a new SMS configuration, you have two options for keeping the currently active SMS configuration information:

- Keep, but never modify, the original SCDS from which the current SMS configuration was activated.

  If you choose this method, you need to maintain a log of all status changes, such as VARY storage group commands, that you make to the currently active SMS configuration. If in the future you activate a different SMS configuration but then decide you want to fall back to your original, you can re-activate the SCDS. You lose all the status changes you have made since activating the SCDS and must re-enter them, but you return to the original SMS configuration.
- Save the current active SMS configuration using the SETSMS operator command:

  ```
  SETSMS SAVEACDS(ACDS.FALLBACK)
  ```

  This is the better alternative.

  ACDS.FALLBACK must be an existing, already allocated data set. By using this command, you not only save the current storage management policy in ACDS.FALLBACK, but you also save the status changes you have made since the original SCDS was activated. You can then activate the new SMS configuration. If in the future you decide you want to fall back to the original SMS configuration, you can use the SETSMS operator command to reactivate it:

  ```
  SETSMS ACDS(ACDS.FALLBACK)
  ```

  This alternative is also useful if you have altered the SCDS that you originally activated.

Note that when you use the SETSMS operator command to change SMS statuses, such as storage group and volume, these changes are not kept if a new configuration is activated. Instead, the MVS statuses override any changes you have made. See *OS/390 MVS System Commands* for information on the SETSMS operator command.

Note: All systems in the SMS complex must be running in the same mode. All systems running releases of DFSMS/MVS or MVS/DFP prior to DFSMS/MVS

1.3 are considered to be running in compatibility (8-name) mode. See "Defining the Base Configuration" on page 27 for more information about compatibility mode.

When an SMS control data set that supports only eight names is accessed for update on a system running in 32-name mode, you must convert the data set to a new, incompatible format in order to support 32 names. Confirm this conversion using the operator console or the ISMF. This conversion is permanent, so you should make copies of your control data sets before the system mode is converted.

If any of the systems are pre-DFSMS/MVS 1.3, then *all* 1.3 or above systems must run in compatibility mode. Be sure to use compatibility mode if you must revert to a lower-level system on failures or on a recovery using ABARS, or if you must ship the control data set to a lower-level system.

OAM Considerations when Changing SCDSs

Activating a new SCDS while OAM is active causes OAM to restart. During this re-initialization, all optical libraries and drives defined to the new SCDS are reset to the initial status values specified in the SCDS.

After the OAM restart completes, display all optical libraries and drives, and tape libraries, and then set them to the desired online/offline status.

| Note that OAM has optional keywords that can override automatic restart of the
| address space. See *DFSMS/MVS OAM Planning, Installation, and Storage
| Administration Guide for Tape Libraries* for more information.

Chapter 11. Maintaining the Storage Management Subsystem

After building a configuration and activating SMS, you need to monitor and adjust it over time. This chapter explains how to maintain SMS after you have activated it. Maintenance activities include listing, altering, copying, and deleting SMS classes, aggregate groups, storage groups, optical libraries and drives, and tape libraries. It also includes listing, altering and ejecting optical and tape volumes.

This chapter discusses the DISPLAY SMS operator command.

Note: You can use the DFSMS/MVS NaviQuest tool to help you maintain your configuration. With DFSMS/MVS NaviQuest, you can perform many ongoing storage administration activities in batch, thereby freeing the workstation for other work. For example, you can update configuration values, create reports, and use NaviQuest's cross-referencing capabilities to help you verify changes to the configuration.

For more information on DFSMS/MVS NaviQuest, see *NaviQuest User's Guide*.

Displaying SMS and OAM Information

You can use the DISPLAY SMS operator command to determine the status of SMS, storage groups, DASD volumes, tape volumes, tape libraries, OAM, OSMC, optical libraries, optical volumes and optical drives.

- VARY SMS—for varying optical drives, optical libraries, or tape libraries online or offline
- DISPLAY SMS—for determining the status of an object or object backup storage group, an optical volume, optical library, optical drive, tape volume, tape library, or tape storage group; as well as for determining OAM, OSMC, OSMC task processing, or OAM XCF status

You can also enter the LISTSYS and LISTVOL line operators on the ISMF Storage Group List panels to get information about the currently active configuration, storage groups and volumes.

See *OS/390 MVS System Commands* for information on using all the operator commands.

For specific information on using the OAM storage management component commands, see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support* and *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Tape Libraries*.

Displaying Information about the Active Configuration

To display information about the currently active configuration, issue the following command:

```
D SMS,ACTIVE or D SMS
```

where D is an abbreviation for DISPLAY. (The default for the DISPLAY command is to display the active configuration when no options are specified.) An example of the generated output appears in Figure 76.

```
09.23.36          IGD002I 09:23:36 DISPLAY SMS 401

SCDS = SMS.SCDS1.SCDS
ACDS = SMS.ACDS1.ACDS
COMMDS = SMS.COMMDS1.COMMDS
DINTERVAL = 150
REVERIFY = NO
ACSDEFAULTS = NO
    SYSTEM     CONFIGURATION LEVEL    INTERVAL SECONDS
    SYSTEM01   1991/12/01 09:23:29          15
    SYSTEM02   ---------- --------         N/A
    SYSTEM03   ---------- --------         N/A
    SYSTEM04   ---------- --------         N/A
    SYSTEM05   ---------- --------         N/A
    SYSTEM06   ---------- --------         N/A
    SYSTEM07   ---------- --------         N/A
    SYSTEM08   ---------- --------         N/A
```

Figure 76. Displaying Information about the Active Configuration

The display shows the names of the current control data sets. The naming convention used here specifies a first level qualifier of SMS for all SMS control data sets. The second level qualifier identifies the type of SMS data set. The third level qualifier uniquely identifies the data set.

The DINTERVAL, REVERIFY ACSDEFAULTS, and OVRD_EXPDT fields contain the values given them when SMS was initialized. For an explanation of these values, see "Initializing SMS through the IGDSMSxx Member" on page 15.

The last portion of the output shows configuration levels about the system from which you issued the DISPLAY command. The configuration level indicates the date and time the ACDS was last updated. This last portion also contains the synchronization time interval (the number of seconds that SMS allows before this system checks the COMMDS for news from other systems in the SMS complex) for each system. The synchronization time interval can be changed using the SET SMS command.

Displaying SMS TRACE Information

To display information about the SMS trace options in effect, issue the command:

```
D SMS,TRACE
```

where D is an abbreviation for DISPLAY. An example of the generated output appears in Figure 77 on page 181.

```
 14.57.24       IGD002I 14:57:24 DISPLAY SMS 879

 TRACE     = {ON|OFF}   SIZE =nnnK     TYPE {ERROR|ALL}
   JOBNAME ={jjj|*}    ASID ={asid|*}
   TRACING EVENTS:
MODULE ={ON|OFF}  SMSSJF ={ON|OFF}  SMSSI  ={ON|OFF}  ACSINT ={ON|OFF}
OPCMD  ={ON|OFF}  CONFC  ={ON|OFF}  CDSC   ={ON|OFF}  CONFS  ={ON|OFF}
MSG    ={ON|OFF}  ERR    ={ON|OFF}  CONFR  ={ON|OFF}  CONFA  ={ON|OFF}
ACSPRO ={ON|OFF}  IDAX   ={ON|OFF}  DISP   ={ON|OFF}  CATG   ={ON|OFF}
VOLREF ={ON|OFF}  SCHEDP ={ON|OFF}  SCHEDS ={ON|OFF}  VTOCL  ={ON|OFF}
VTOCD  ={ON|OFF}  VTOCR  ={ON|OFF}  VTOCC  ={ON|OFF}  VTOCA  ={ON|OFF}
RCD    ={ON|OFF}  DCF    ={ON|OFF}  DPN    ={ON|OFF}  TVR    ={ON|OFF}
DSTACK ={ON|OFF}
```

Figure 77. Displaying Trace Information

The display shows the status of the SMS trace option, the size of the SMS trace table, and the type of SMS trace entries. ERR means only error type trace entries are traced. ALL indicates all types of trace entries are traced. This section also includes the JOBNAME which indicates the tracing scope in relation to jobs being run. ASID indicates the tracing scope in relation to address spaces. ASID means tracing is limited to a particular address space and * indicates tracing is performed for all address spaces.

The last section, TRACING EVENTS, indicates which SMS events are selected for tracing.

You can use this tracing display if you have been requested to collect information by the IBM Support Center for diagnosing problems. See *DFSMS/MVS DFSMSdfp Diagnosis Reference*for additional information.

Displaying Storage Group Status Using the DISPLAY SMS Command

You can use the STORGRP parameter to display the status of storage groups including tape and object storage groups. Figure 78 shows the syntax of the DISPLAY SMS,STORGRP command. See *OS/390 MVS System Commands* for the complete syntax of the DISPLAY SMS command.

|

```
{DISPLAY} SMS{,STORGRP(STORGRP)} [LISTVOL|DETAIL][,L={a}]
{D}          {,SG     (ALL)}                     [   {cc}]
                                                 [   {cca}]
                                                 [   {name}]
                                                 [   {name-a}]
```

Figure 78. DISPLAY SMS,STORGRP Command Syntax

| **STORGRP({storgrp|ALL})**

If STORGRP(storgrp) is specified, the system displays the status of one storage group for each MVS system connected to that storage group. If STORGRP(ALL) is specified, the system displays the status of all the storage groups defined in the SMS configuration. You can abbreviate STORGRP as SG.

LISTVOL

displays the status and volume serial numbers of all the volumes in the storage group. The LISTVOL parameter is ignored for tape, object, and object backup storage groups. This parameter is mutually exclusive with the DETAIL parameter.

DETAIL
displays detail status and is only valid for tape, object, and object backup storage groups. This parameter is mutually exclusive with the LISTVOL parameter and overrides the LISTVOL parameter if the storage group is tape or object related.

L=a|cc|cca|name|name-a
specifies where the results of the inquiry are to be displayed: the display area (a), the console (cc), or both (cca). The name parameter is routed to the console referred to by ″name″ and the name-a parameter is routed to the console referred to by ″name″ and the screen referred to by ″a″. The name parameter can be an alphanumeric character string.

To display information about the status of a storage group, issue the following command:

```
D SMS,STORGRP(storgrp)
```

For `storgrp` you specify the name of a single storage group or you specify `ALL` to display the status of all storage groups. An example of the generated output for the single storage group SG1 appears in Figure 79.

```
15.12.25         d sms,storgrp(s1p01)
15.12.25         IGD002I 15:12:25 DISPLAY SMS 900
STORGRP TYPE              SYSTEM= 1 2 3 4 5 6 7 8
SG1     {DUMMY|POOL|TAPE|VIO}     + . . . . . . .
****************************** LEGEND ******************************
. THE STORAGE GROUP OR VOLUME IS NOT DEFINED TO THE SYSTEM
+ THE STORAGE GROUP OR VOLUME IS ENABLED
- THE STORAGE GROUP OR VOLUME IS DISABLED
* THE STORAGE GROUP OR VOLUME IS QUIESCED
D THE STORAGE GROUP OR VOLUME IS DISABLED FOR NEW ALLOCATIONS ONLY
Q THE STORAGE GROUP OR VOLUME IS QUIESCED FOR NEW ALLOCATIONS ONLY
SYSTEM 1 = sysname1    SYSTEM 2 = sysname2     SYSTEM 3 = sysname3
SYSTEM 4 = sysname4    SYSTEM 5 = sysname5     SYSTEM 6 = sysname6
SYSTEM 7 = sysname7    SYSTEM 8 = sysname8     SYSTEM 9 = sysname8
SYSTEM 10= sysname10   SYSTEM 11= sysname11    SYSTEM 12= sysname12
SYSTEM 13= sysname13   SYSTEM 14= sysname14    SYSTEM 15= sysname15
SYSTEM 16= sysname16   SYSTEM 17= sysname17    SYSTEM 18= sysname18
SYSTEM 19= sysname19   SYSTEM 20= sysname20    SYSTEM 21= sysname21
SYSTEM 22= sysname22   SYSTEM 23= sysname23    SYSTEM 24= sysname24
SYSTEM 25= sysname25   SYSTEM 26= sysname26    SYSTEM 27= sysname27
SYSTEM 28= sysname28   SYSTEM 29= sysname29    SYSTEM 30= sysname30
SYSTEM 31= sysname31   SYSTEM 32= sysname32
```

Figure 79. Displaying Storage Group Status Information

Note that if you have more than 16 systems in your SMS complex, a second message follows, and then the legend follows that.

Displaying the Status of Volumes in the Storage Group

Specifying the optional LISTVOL parameter provides the status of the volumes associated with the storage group. If you have volume 123456 in storage group SG1, the following command generates output similar to that shown in Figure 80 on page 183.

```
D SMS,STORGRP(SG1),LISTVOL
```

```
15.31.25           IGD002I 15:31:25 DISPLAY SMS 912
STORGRP TYPE               SYSTEM= 1 2 3 4 5 6 7 8
SG1     {DUMMY|POOL|TAPE|VIO}       + . . . . . . .
VOLUME   UNIT    SYSTEM= 1 2 3 4 5 6 7 8 STORGRP NAME
123456                  + . . . . . . .   SG1
**************************** LEGEND ****************************
. THE STORAGE GROUP OR VOLUME IS NOT DEFINED TO THE SYSTEM
+ THE STORAGE GROUP OR VOLUME IS ENABLED
- THE STORAGE GROUP OR VOLUME IS DISABLED
* THE STORAGE GROUP OR VOLUME IS QUIESCED
D THE STORAGE GROUP OR VOLUME IS DISABLED FOR NEW ALLOCATIONS ONLY
Q THE STORAGE GROUP OR VOLUME IS QUIESCED FOR NEW ALLOCATIONS ONLY
SYSTEM 1 = sysname1    SYSTEM 2 = sysname2     SYSTEM 3 = sysname3
SYSTEM 4 = sysname4    SYSTEM 5 = sysname5     SYSTEM 6 = sysname6
SYSTEM 7 = sysname7    SYSTEM 8 = sysname8     SYSTEM 9 = sysname8
SYSTEM 10= sysname10   SYSTEM 11= sysname11    SYSTEM 12= sysname12
SYSTEM 13= sysname13   SYSTEM 14= sysname14    SYSTEM 15= sysname15
SYSTEM 16= sysname16   SYSTEM 17= sysname17    SYSTEM 18= sysname18
SYSTEM 19= sysname19   SYSTEM 20= sysname20    SYSTEM 21= sysname21
SYSTEM 22= sysname22   SYSTEM 23= sysname23    SYSTEM 24= sysname24
SYSTEM 25= sysname25   SYSTEM 26= sysname26    SYSTEM 27= sysname27
SYSTEM 28= sysname28   SYSTEM 29= sysname29    SYSTEM 30= sysname30
SYSTEM 31= sysname31   SYSTEM 32= sysname32
```

Figure 80. Displaying Storage Group Volume Status Information

Displaying Storage Group Detail Status

If you specify STORGRP with a single storage group name, you get detailed information about the optical libraries in the requested object or object backup storage group. You could also get information about the tape libraries in the requested tape storage group.

For sample command and output examples, see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support* and *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Tape Libraries*.

Displaying OAM Status in a Parallel Sysplex

Specifying the OAMXCF parameter displays the status of this instance of OAM within a Parallel Sysplex. You get the status of each member within the specific Parallel Sysplex, as well as the number of transactions waiting for a response from other instances of OAM in the Parallel Sysplex.

```
D SMS,OAMXCF
```

An example of the generated output appears below.

```
CBR1250I OAM XCF status&colon;
XCF MEMBER NAME      USER          SYSTEM       OPT        OPT     TAPE
                                         STATE         NAME
READ    WRITE    READ
OAMSYS1                        ACTIVE        SYSTEM2
0             0             0
OAMSYS2                        ACTIVE        SYSTEM3
0             0             0
------------------------------------------------------------
OAMSYS1                        ACTIVE        SYSTEM1          0
           0             0
XCF GROUP NAME:  OAMGRP1
```

Figure 81. Displaying OAM Status in a Parallel Sysplex

| For more information on object support, see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support* and *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Tape Libraries*.

Displaying the Caching Statistics

| If you have an IBM 3990 Storage Control with cache with at least one SMS volume attached, you can display cache statistics, hit ratio, and DASD Fast Write bypasses, by issuing the following command:

```
D SMS,CACHE
```

| An example of the generated output appears below.

```
IGD002I 09:28:38 DISPLAY SMS 411
  SSID  DEVS  READ  WRITE  HIT RATIO  FW BYPASSES
  0002   32    80    60        68%    62
  0004   14    92   100        76%    30
  000E   20    84   100        84%    44
  0011   08   100   100        92%    16
  001C   08   100   100        86%    24
***************************** LEGEND *****************************
SSID        THE SUBSYSTEM IDENTIFIER OF THE IBM 3990 STORAGE CONTROL
            MODEL 3 TO WHICH THE INFORMATION APPLIES
DEVS        THE NUMBER OF SMS-MANAGED DEVICES ATTACHED TO THE IBM 3990
            STORAGE CONTROL MODEL 3 WITH THIS SSID
READ        THE PERCENTAGE OF CYLINDERS FOR WHICH DYNAMIC CACHING IS
            PERMITTED
WRITE       THE PERCENTAGE OF CYLINDERS FOR WHICH DYNAMIC DASD FAST
            WRITE IS PERMITTED
HIT RATIO   THE READ HIT RATIO THAT IS ACHIEVED BY THE IBM 3990 STORAGE
            CONTROL MODEL 3 WITH THIS SSID
FW BYPASSES THE NUMBER DASD FAST WRITES PER MINUTE THAT ARE EXECUTED
            DIRECTLY WITH DASD BECAUSE OF MVS OVER-COMMITMENT
```

Figure 82. Displaying Cache and DASD Fast Write Information

Note: The read and write percentages is displayed as N/A if the subsystem has the extended platform available. Since the enhanced dynamic cache management algorithm is used, the percentages are no longer valid.

Displaying Storage Group Status Using ISMF

You can display a storage group’s status in each system in the SMS complex by using the LISTSYS line operator in ISMF. The storage group must be a pool type, and it must be part of the active configuration. Go to the Storage Group Application Selection panel, shown in Figure 13 on page 45, from the ISMF Primary Option Menu. Then create a list of storage groups. For example, specifying the following information on the Storage Group Application Selection panel creates a list of storage groups whose names begin with SGNAME:

```
CDS NAME            ===> 'ACTIVE'
STORAGE GROUP NAME  ===> SGNAME*
SELECT OPTION       ===> 1
```

Press the ENTER key to display the Storage Group List panel, and enter LISTSYS in the Line Operator column next to the desired pool type storage group. Press ENTER to display the Storage Group System Status Display panel shown in Figure 83.

Note: If a volume has been initialized with LABEL as an SMS device, the storage group must be defined in the active configuration in order for space information to be displayed when the LISTSYS line operator is issued.

```
  Control  Utilities  Scroll  Help
------------------------------------------------------------------------------
                       SMS STORAGE GROUP STATUS DISPLAY           Page 1 of 2
Command ===>

CDS Name . . . . . : ACTIVE
Storage Group Name : SGNAME01           * SYS GROUP = sysplex minus systems in
Total Space for Storage Group             sysplex explicitly defined in SCDS
 Mb-total: -------
 Mb-free : -------   Available Group                     Available Group
 % Free  :  --    Space For Allocation                 Space For Allocation
System/Sys SMS SG Mb-     Mb-     %     System/Sys SMS SG Mb-     Mb-     %
Group Name Status Total   Free    Free  Group Name Status Total   Free    Free
---------- ------ ------- ------- ----  ---------- ------ ------- ------- ----
*SYSPX1     ENABLE ------- -------   --  SYSTEM2    NOTCON ------- -------   --
*SYSPX2     NOTCON ------- -------   --  SYSTEM4    NOTCON ------- -------   --
 SYSTEM5    NOTCON ------- -------   --  SYSTEM6    NOTCON ------- -------   --
 SYSTEM7    NOTCON ------- -------   --  SYSTEM8    NOTCON ------- -------   --




Use HELP Command for Help; Use DOWN Command to View Next Panel; END to Exit.
```

Figure 83. Displaying Storage Group Status for Each System/System Group

The Total Space for Storage Group value is the total capacity of all the online volumes belonging to storage group SGNAME01, of all the volumes that have a status of ENABLE.

MB-TOTAL
is the total number of MB belonging to the storage group.

MB-FREE
is the total number of free MB belonging to the storage group.

% FREE
is the percentage of free storage group space.

Note: Space information exists for a volume only if the volume has at least one SMS-managed data set allocated on it.

The Group Space Available for Allocation value represents the sum of all volume space that is currently available to each system online to MVS. The fields contain values only if the SMS SG Status value is ENABLE.

MB-TOTAL
is the total available space in MB.

MB-FREE
is the total available space in MB that is free.

% FREE
is the percentage of total available space that is free.

Displaying Volume Status Using ISMF

You can display a volume's status by going to the Storage Group Application Selection panel, shown in Figure 13 on page 45 from the ISMF Primary Option Menu. Select option 4, VOLUME, and the Storage Group Selection panel is displayed. After entering the single volume or range of volumes you wish to display, choose option 1 (Display) to access the SMS Volume Status Display panel shown in Figure 84.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                          SMS VOLUME STATUS DISPLAY              Page 1 of 2
Command ===>

CDS Name . . . . . . . : ACTIVE
Storage Group Name . . : S1P01
Volume Serial Number . : 1P0101
Physical Volume Status : CONVERTED        * SYS GROUP = sysplex minus systems
               Mb-free : -------            in the sysplex explicitly defined
               % Free  : --                 in the SCDS

  System/Sys SMS Vol MVS Vol  SMS SG  |  System/Sys SMS Vol MVS Vol  SMS SG
  Group Name Status  Status   Status  |  Group Name Status  Status   Status
  ---------- ------- -------- ------  |  ---------- ------- -------- ------
  SYSTEM1    ENABLE  ONLINE   ENABLE  |  SYSTEM2    ENABLE  -------- NOTCON
  SYSTEM3    ENABLE  -------- NOTCON  |  SYSTEM4    NOTCON           NOTCON
  SYSTEM5    NOTCON           NOTCON  |  SYSTEM6    NOTCON           NOTCON
  SYSTEM7    NOTCON           NOTCON  |  SYSTEM8    NOTCON           NOTCON
                                      |
                                      |
                                      |
Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 84. Displaying the SMS Volume Status

The Physical Volume Status field shows the current SMS status of, and the amount of free space on, the volume. The volume status can be:

INITIAL
The volume has been defined to a storage group, but all the data sets are not yet under SMS control.

CONVERTED
The volume is fully converted to SMS.

NONSMS
The volume is not under SMS control.

UNKNOWN
The status of the volume is not known.

The Mb-free value is the total number of free megabytes belonging to the volume.

The % Free value is the percentage of free volume space.

Note: If you do not specify 'ACTIVE' in the CDS Name field, the Physical Volume Status, Mb-free, % Free and MVS Vol Status fields display dashes. The space information only exists for a volume if the volume has at least one SMS-managed data set allocated on it.

The SMS Vol Status field shows the relationship between the volume and SMS. The six possible relationships are:

DISALL
SMS does not permit data sets on this volume to be accessed.

DISNEW
SMS does not permit the allocation of new data sets on this volume.

ENABLE
SMS permits access to data sets on this volume.

NOTCON
SMS does not attempt to access this volume.

QUIALL
SMS does not schedule any more jobs that access data sets on this volume.

QUINEW
SMS does not schedule any jobs that create new data sets on this volume.

The MVS Vol Status column shows relationships between MVS and the volume whose status is being displayed. Possible relationships are:

BOXED
A status encountered when an application cannot disconnect from a malfunctioning volume. BOXED means that MVS is simulating I/O errors in response to the application's I/O request.

NOTREADY
The volume cannot send nor receive I/O now.

OFFLINE
I/O is not possible because the storage management subsystem cannot find the address where the volume's VOLSER is mounted.

ONLINE
The volume is physically connected to MVS; I/O can proceed.

PENDOFF
(Pending Offline) MVS varies the volume offline as soon as persons currently accessing data sets on it have disconnected from it.

Displaying Volumes Using the DISPLAY SMS Command

To display the status of specific DASD, tape or optical disk volumes issue the following command:

```
D SMS,VOLUME(serial-number)
```

If the volume serial number is of a DASD volume in a pool storage group, you get the generated output on the Operator Display panel shown in Figure 85. In this example, the volume serial number is 123456.

```
15.38.00          IGD002I 15:38:00 DISPLAY SMS 918
VOLUME   UNIT    SYSTEM= 1 2 3 4 5 6 7 8 STORGRP NAME
123456   123             + . . . . . . .   SG1
***************************** LEGEND *****************************
. THE STORAGE GROUP OR VOLUME IS NOT DEFINED TO THE SYSTEM
+ THE STORAGE GROUP OR VOLUME IS ENABLED
- THE STORAGE GROUP OR VOLUME IS DISABLED
* THE STORAGE GROUP OR VOLUME IS QUIESCED
D THE STORAGE GROUP OR VOLUME IS DISABLED FOR NEW ALLOCATIONS ONLY
Q THE STORAGE GROUP OR VOLUME IS QUIESCED FOR NEW ALLOCATIONS ONLY
SYSTEM 1 = sysname1    SYSTEM 2 = sysname2     SYSTEM 3 = sysname3
SYSTEM 4 = sysname4    SYSTEM 5 = sysname5     SYSTEM 6 = sysname6
SYSTEM 7 = sysname7    SYSTEM 8 = sysname8     SYSTEM 9 = sysname8
SYSTEM 10= sysname10   SYSTEM 11= sysname11    SYSTEM 12= sysname12
SYSTEM 13= sysname13   SYSTEM 14= sysname14    SYSTEM 15= sysname15
SYSTEM 16= sysname16   SYSTEM 17= sysname17    SYSTEM 18= sysname18
SYSTEM 19= sysname19   SYSTEM 20= sysname20    SYSTEM 21= sysname21
SYSTEM 22= sysname22   SYSTEM 23= sysname23    SYSTEM 24= sysname24
SYSTEM 25= sysname25   SYSTEM 26= sysname26    SYSTEM 27= sysname27
SYSTEM 28= sysname28   SYSTEM 29= sysname29    SYSTEM 30= sysname30
SYSTEM 31= sysname31   SYSTEM 32= sysname32
```

Figure 85. Displaying Status Information for Individual DASD Volumes

If the volume serial number is of an optical volume in an object storage group, you get the generated output on the Operator Display panel shown in Figure 86.

```
CBR1140I OAM volume status: 906
VOLUME STORAGE  RD WR WP MEDIA    FREE SPACE    MOUNTED  PENDING   REQ
       GROUP             TYPE       (KB)   (%)  DRIVE    MOUNT     CT
OBJ4KA *SCRTCH* Y  Y  N  3995-2RW   559376  87% P156AD1  --------    0
OBJ4KB *SCRTCH* Y  Y  N  3995-2RW   594356  93%          --------    0
3995 (1300 MB) rewritable optical disk media.
LIBRARY: P156A
PSEUDO LIBRARY:
OWNER: LCS_TEST
XCF MEMBER NAME: OAMSYS1
VOLSER:            - OBJ4KA -  - OBJ4KB -
CREATION DATE:     1998-07-23  1998-07-23
LAST WRITTEN DATE: 1998-07-23  1998-07-23
LAST MOUNTED DATE: 1998-07-23  1998-07-23
ENTER-EJECT DATE:  1998-07-23  1998-07-23
EXPIRATION DATE:   0001-01-01  0001-01-01
```

Figure 86. Displaying Status Information for Individual Optical Volumes

If the volume serial number is of a tape volume, you get the generated output on the Operator Display panel shown in Figure 87 on page 189.

```
CBR1180I OAM tape volume status: 147
VOLUME  MEDIA     STORAGE   LIBRARY   USE  W  C  SOFTWARE  LIBRARY
        TYPE      GROUP     NAME      ATR  P  P  ERR STAT  CATEGORY
RTS997  MEDIA2    SGMIXED   ATLF4007  P    N  N  NOERROR   NOTAVAIL
-------------------------------------------------------------------
RECORDING TECH:     36 TRACK
COMPACTION:         YES
SPECIAL ATTRIBUTE:  NONE
CREATION DATE:      1998-08-17       EXPIRATION DATE:    2025-04-07
LAST MOUNTED DATE:  1998-08-10       LAST WRITTEN DATE:  1998-08-10
ENTER/EJECT DATE:   1998-04-07
SHELF LOCATION:     OUT
OWNER: LCS_TESTER
----------------------------------------------------------------
```

Figure 87. Displaying the Status of a Tape Volume

Displaying the Status of a Device

You can use the DEVSERV command to display the status of a device.

```
{DEVSERV} {SMS},ddd[,nn][,ONLINE][,L={a}]
{DS}      {S}          [,ON]     [   {cc}]
                                 [   {cca}]
                                 [   {name}]
                                 [   {name-a}]
```

Figure 88. DEVSERV Command Syntax

ddd specifies the device number, in hexadecimal, for which the system is to display information.

nn specifies the decimal number (from 1 to 32) of devices for which the system is to display the information, in ascending order beginning with the device you specify. If you do not code *nn*, the system displays information about the one device you specify.

ONLINE/ON
displays information about only those specified devices that are online. If you do not specify ONLINE or OFFLINE, the system displays information about both online and offline devices.

OFFLINE/OFF
displays information about only those specified devices that are offline.

L=a|cc|cca|name|name-a
specifies where the results of the inquiry are to be displayed: the display area (a), the console (cc), or both (cca). The name parameter is routed to the console referred to by ″name″ and the name-a parameter is routed to the console referred to by ″name″ and the screen referred to by ″a″.

The command `DEVSERV S,430` produces the output shown in Figure 89 on page 190 for target device 430.

```
15.44.26          IGD001I 15:44:26 DEVSERV SMS 921
 UNIT,DTYPE   ,M,VOLSER,VOLSTAT    STORGRP,SGSTAT
 430 ,3380    ,O,XP0101,ENABLED    SXP01,QUIESCED

*******************************LEGEND*******************************
A = ALLOCATED                      F = OFFLINE
M = MOUNT PENDING                  N = NOT ALLOCATED
O = ONLINE                         P = PENDING OFFLINE
```

Figure 89. Displaying the Status of a Device

See *OS/390 MVS System Commands* for the complete syntax of the DEVSERV command.

Listing SMS Classes, Aggregate Groups, Storage Groups, and Libraries Using ISMF

You can list SMS classes, aggregate groups, storage groups, optical libraries, optical drives, saved lists, DASD volumes, optical volumes, tape libraries, tape volumes, and data sets on application panels. You can also generate listings by selecting the LIST option on application panels or by typing the LIST line operator. "Appendix A. ISMF Command and Line Operator Reference Summary" on page 287 summarizes all the ISMF commands and line operators, describes them, and lists the applications from which you can issue them.

ISMF supports the following screen sizes for various lists:

24 x 80

27 x 132

32 x 80

43 x 80

31 x 160

The examples shown in this book are for displays with a screen size of 24 x 80.

Note: 31 x 160 is half of a 3290 screen. Only half of the 3290 screen is used.

Listing with View and Sort

When you specify the list option on the Data Class, Management Class, Storage Class, Storage Group, Aggregate Group, Library, or Drive Application Selection panels, the values you specify in the Respecify View Criteria field and the Respecify Sort Criteria field determine characteristics of the list that ISMF displays.

If you specify yes, Y, in the Respecify View Criteria field of the Application Selection panel, you see the View Entry panel for that application. You can specify which data columns appear in the list and the order in which they appear, and you can save your specifications to use again. If you specify no, N, in the Respecify View Criteria field, ISMF uses the last used values for data column selection and order. The initial value is all data columns in their default order.

If you specify yes, Y, in the Respecify Sort Criteria field of the Application Selection panel, you see the Sort Entry panel for that application. You can specify by which data columns entries in the list are sorted. You can only specify data columns that are already specified as view criteria. If you specify no, N, in the Respecify Sort Criteria field, ISMF uses the default sorting order.

“Listing Data Classes” gives an example of how the View and Sort panels work together. You can find detailed information about View and Sort in *DFSMS/MVS Using ISMF*.

Listing Data Classes

You can use View and Sort to customize the way you see a list. For example, if you want to see a list of data classes with their Data Set Name Type, Logical Record Length, and Record Organization data columns displayed, and you want the entries to appear sorted by data set name type in ascending order, you can follow the panels in this section.

Choose the LIST option from the Data Class Application Selection panel, shown in Figure 44 on page 116, and select the viewing order by specifying Y, yes, in the Respecify View Criteria field. Select the sorting order by specifying Y, yes, in the Respecify Sort Criteria field. ISMF displays the Data Class View Entry panel. If you want to see Data Set Name Type, Logical Record Length, and Record Organization, type their corresponding tags, separated by blanks, in the field called Specify Tags in Sequence Desired, as shown in Figure 90 on page 192. Your selections are displayed in the order that you typed them.

Line Operator and Data Class Name always appear as the first and second data columns in a list but, because you cannot select them from this panel, they do not have corresponding tags. You cannot specify them on the this panel as selected tags.

You can choose option 2, Save, to save your specifications for later use.

Figure 90 on page 192 shows page 1 of the Data Class View Entry Panel.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                        DATA CLASS VIEW ENTRY PANEL              Page 1 of 2
Command ===>

If desired, Select option . .           (1 - SELECT, 2 - SAVE, 3 - DELETE)
                View Name . .           (1 to 8 characters, * to Delete all,
                                          or blank for List of Saved Views)
Specify tags in Sequence Desired:
===> *

       Line Operator                    (16)  Imbed
       Data Class Name                  (32)  Initial Load
 (15)  Additional Volume Amt             (6)  Key Length
  (8)  Avgrec                            (7)  Key Offset
  (9)  Avg Value                        (24)  Last Date Modified
 (34)  BWO                              (23)  Last Modified Userid
 (28)  Compaction                       (25)  Last Time Modified
 (18)  Control Interval Size Data       (35)  Log
 (26)  Data Set Name Type                (5)  Logical Record Length
 (27)  Extended Addressability          (36)  Logstream Id

Use ENTER to Perform Selection or Display List in VIEW Sequence;
Use DOWN for next Panel; Use HELP Command for Help; Use END Command to Exit.
```

Figure 90. Specifying Data Class View Criteria, Page 1 of 2

Figure 91 shows page 2 of the Data Class View Entry Panel.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                        DATA CLASS VIEW ENTRY PANEL              Page 2 of 2
Command ===>

If desired, Select option . .           (1 - SELECT, 2 - SAVE, 3 - DELETE)
                View Name . .           (1 to 8 characters, * to Delete all,
                                          or blank for List of Saved Views)
Specify tags in Sequence Desired:
===> *

 (29)  Media type                       (31)  Reuse
 (19)  % Freespace Control Area         (21)  Shareoptions Crossregion
 (20)  % Freespace Control Interval     (22)  Shareoptions Crosssystem
 (30)  Recording Technology             (37)  Space Constraint Relief
 (39)  Record Access Bias               (12)  Space Directory
  (4)  Record Format                    (10)  Space Primary
  (3)  Record Organization              (11)  Space Secondary
 (38)  Reduce Space Up To (%)           (33)  Spanned / Nonspanned
 (17)  Replicate                        (14)  Volume Count
 (13)  Retpd Or Expdt

Use ENTER to Perform Selection or Display List in VIEW Sequence;
Use UP For previous Panel; Use HELP Command for Help; Use END Command to Exit.
```

Figure 91. Specifying Data Class View Criteria, Page 2 of 2

Next, ISMF displays the Data Class Sort Entry panels. Enter 25 as the Major Field Sequence and A as the Major Field Order as shown in Figure 92 on page 193 and Figure 93 on page 193 to sort in ascending order by Data Set Name Type. Tags for Line Operator and Data Class Name always appear on this panel. Notice that the only other tags displayed are the ones you specified in the Data Class View Entry panel.

Figure 92 shows page 1 of the Data Class Sort Entry Panel.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                       DATA CLASS SORT ENTRY PANEL                Page 1 of 2
Command ===>

Specify one or more Attribute Numbers for Sort Sequence:
  Major Field  . . . 2    Minor Field 1 . . .      Minor Field 2 . . .

Specify A for Ascending or D for Descending Sort Order:
  Major Field  . . . A    Minor Field 1 . . .      Minor Field 2 . . .

  (1)  Line Operator                (16)  Imbed
  (2)  Data Class Name              (32)  Initial Load
 (15)  Additional Volume Amt         (6)  Keylength
  (8)  Avgrec                        (7)  Key Offset
  (9)  Avg Value                    (24)  Last Date Modified
 (34)  BWO                          (23)  Last Modified Userid
 (28)  Compaction                   (25)  Last Time Modified
 (18)  Control Interval Size data   (35)  Log
 (26)  Dataset Name Type             (5)  Logical Record Length
 (27)  Extended Addressability      (36)  Logstream Id

Use ENTER to Perform SORT; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 92. Specifying Data Class Sort Criteria, Page 1 of 2

Figure 93 shows page 2 of the Data Class Sort Entry Panel.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                       DATA CLASS SORT ENTRY PANEL                Page 2 of 2
Command ===>

Specify one or more Attribute Numbers for Sort Sequence:
  Major Field  . . . 2    Minor Field 1 . . .      Minor Field 2 . . .

Specify A for Ascending or D for Descending Sort Order:
  Major Field  . . . A    Minor Field 1 . . .      Minor Field 2 . . .

 (29)  Media Type                    (31)  Reuse
 (19)  % Freespace Control Area      (21)  Shareoptions Crossregion
 (20)  % Freespace Control Interval  (22)  Shareoptions Crosssystem
 (39)  Record Access Bias            (37)  Space Constraint Relief
  (4)  Record Format                 (12)  Space Directory
  (3)  Record Organization           (10)  Space Primary
 (30)  Recording Technology          (11)  Space Secondary
 (38)  Reduce Space Up To (%)        (33)  Spanned / Nonspanned
 (17)  Replicate                     (14)  Volume Count
 (13)  Retpd or Expdt

Use ENTER to Perform SORT; Use UP Command to View previous Panel;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 93. Specifying Data Class Sort Criteria, Page 2 of 2

On the basis of the view and sort criteria, ISMF displays a list of data classes like the one shown in Figure 94 on page 194. On the listing, you can enter line operators such as COPY, DISPLAY, and HIDE against individual entries. Some line operators, such as ALTER and DELETE, cannot be used with the ACTIVE control data set.

```
  Panel  List  Utilities  Scroll  Help
------------------------------------------------------------------------------
                                DATA CLASS LIST
Command ===>                                                  Scroll ===> HALF
                                                              Entries 1-8 of 8
                                                              View in Use
CDS Name . : SCDS.DA

Enter Line Operators below:

     LINE       DATACLAS
     OPERATOR   NAME     DATA SET NAME TYPE  LRECL  RECORG
    ---(1)----  --(2)--- -------(26)-------  -(5)-  -(3)--
                DA000000 ------------------     80  --
                DA000003 ------------------    800  --
                DA000004 ------------------   2550  --
                DA000005 ------------------     80  --
                DA000006 PDS                    80  --
                DA000007 EXTENDED REQUIRED    ----  --
                DA000008 ------------------    255  --
                DA000009 ------------------  32760  --
    ----------  -------- ------  BOTTOM  OF  DATA  ------  --------  ----------
```

Figure 94. Displaying a List of Data Classes

You can scroll up, down, left, and right. However, ISMF always displays the LINE OPERATOR and DATACLAS NAME fields on the left side of the panel. You can also change the View and Sort criteria by issuing the View or Sort command on the command line.

Listing Storage Groups, Management Classes, Storage Classes, Aggregate Groups, Libraries and Drives

The viewing, sorting and saving procedures for lists of storage groups, storage classes, and libraries correspond to the example of the procedures for data classes.

Processing Aggregate Groups

With Aggregate Backup and Recovery Support (ABARS II), new attributes were added and other attributes were deleted. Therefore, when a storage administrator processes aggregate groups, different results are obtained depending on the ISMF version installed.

When an aggregate group created on a system prior to DFSMS/MVS 1.1 is processed on a DFSMS/MVS system the following occurs:

- The deleted attributes (Expiration Date, Destination, and Tolerate Enqueue Failure) do not appear on the List, Display or Alter panels.
- The added attributes (Management Class Name and Number of Copies) appear as blanks on the Display and Alter panels, and hyphens on the List panel.

If an aggregate group created on a system prior to DFSMS/MVS 1.1 is specified on the Aggregate Display or Alter panel, the compatibility message (DGTAG031) is displayed, and the message help panel (DGTMAG31) contains the values defined in the aggregate group. This compatibility warning is not displayed when the storage administrator converts the aggregate group to a DFSMS/MVS aggregate group (by specifying the Number of Copies field).

When a DFSMS/MVS aggregate group is processed on a system prior to DFSMS/MVS 1.1, the following occurs:

- The attributes deleted in DFSMS/MVS (Expiration Date, Destination, and Tolerate Enqueue Failure) appear as blanks on the List, Display or Alter panels, and hyphens on the List panel until they are defined in the construct.
- Because the attributes deleted in DFSMS/MVS are all required fields in the system prior to DFSMS/MVS 1.1, the construct is not usable until it is altered to define those fields, and you cannot exit the Aggregate Group Alter panel until they are defined (unless the CANCEL command is used).
- The attributes added in DFSMS/MVS (Management Class Name and Number of Copies) do not appear on any of the Aggregate Group panels.

Processing Management Classes

With the introduction of ABARS II the number of columns on the Management Class list was modified to include columns 31 through 38. Even though the number of attributes has increased, the Display, Define and Alter functions are simplified with the reorganization of the current and new management class attributes into four separate attribute groups: Space, Backup, Aggregate Backup, and Class Transition. When displaying, defining, or altering a management class, you can specify which set of attributes to process first.

When the storage administrator processes a pre-DFSMS/MVS 1.1 management class construct on an DFSMS/MVS system, the following occurs:

- The added attributes appear as blanks on the Display and Alter panels, and hyphens on the List panel.
- The construct does not need to be modified, since all added attributes are optional.

When a DFSMS/MVS management class construct is processed in a pre-DFSMS/MVS Version 1 Release 1 system, the added attributes do not appear on any of the Management Class panels.

Altering Data Set Associations

You can change the management class or storage class associated with a data set. The management class or storage class you choose for the data set must be in the ACDS. Enter the ALTER line operator against a data set entry on a Data Set List (option 1 in the ISMF Primary Option Menu).

Altering SMS Classes, Aggregate Groups, Storage Groups, Libraries, Volumes, and Drives

You can change definitions of SMS classes, storage groups, volumes, aggregate groups, optical libraries, optical drives, and tape libraries by selecting the alter option from the Application Selection panels. You can also change the definitions by entering the ALTER line operator against a list of constructs. This method is more efficient if you want to make multiple changes in one SCDS. For information on altering optical libraries, see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support*. For information on altering tape libraries and drives, see *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Tape Libraries*.

Note: If you are in user mode, you are not authorized to alter SMS classes and libraries. You must be in storage administrator mode to be able to alter SMS classes.

Altering Storage Groups

As the needs of your installation change, you might need to modify storage groups. To modify a storage group, go to the Storage Group Application Selection panel, shown in Figure 13 on page 45, and specify the name of the SCDS containing the storage group you want to modify. Specify the storage group name and select option 3, ALTER. ISMF displays the Storage Group Alter panel.

| You can also use the SETSMS operator command to change the SMS status of storage groups. Note, however, that SMS statuses changed in this way are overridden by MVS statuses if a new configuration is activated.

Redefining Volumes among Storage Groups

If you have defined DASD volumes to belong to one storage group but now want them to belong to a different storage group, go to the Storage Group Application Selection panel. Specify the CDS name and storage group name, then select option 4, Volume. Press ENTER to get the Storage Group Volume Selection panel shown in Figure 95.

In the Specify a Single Volume (in Prefix), or Range of Volumes field, enter the

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                        STORAGE GROUP VOLUME SELECTION
Command ===>

CDS Name  . . . . . : SMS.SCDS1.SCDS
Storage Group Name  : PRIMARY
Storage Group Type  : POOL

Select One of the following Options:
  4  1. Display      - Display SMS Volume Statuses (Pool only)
     2. Define       - Add Volumes to Volume Serial Number List
     3. Alter        - Alter Volume Statuses   (Pool only)
     4. Delete       - Delete Volumes from Volume Serial Number List

Specify a Single Volume (in Prefix), or Range of Volumes:
      Prefix   From     To    Suffix  Hex
      ______  ______  ______  _____    _
 ===> SYS     001     077                    ('X' in HEX field allows
 ===> DFPIP1                                  FROM - TO range to include
 ===> SY      25      30      A        X      hex values A through F.)
 ===>         001     010     VOL
Use ENTER to Perform Selection;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 95. Moving a Volume from One Storage Group to Another

serial numbers of the volumes you want to redefine, select option 4 to delete them from their current storage group, and press ENTER. Then, return to the Storage Group Application Selection panel, enter the name of the new storage group to contain the volumes, and select option 4, Volume, which returns you to the Storage Group Volume Selection panel. Specify the volume serial numbers, select option 2 to define them to the storage group, and press ENTER. You need to activate the updated configuration.

You are responsible for maintaining consistent physical connections and being aware of multivolume data sets. All of a multivolume data set or a VSAM sphere (the base cluster and its associated alternate indexes) must be in a single storage group. You must also maintain storage group, management class, and storage class consistency. If a data set has a management class with certain migration and backup attributes, and you redefine the volume containing that data set from one storage group to another, the new storage group might not be eligible for the same migration and backup processing.

You can also use the SETSMS operator command to change the SMS status of volumes. Note, however, that SMS statuses changed in this way are overridden by MVS statuses if a new configuration is activated.

Altering Management Classes

As the needs of your installation change, you might need to modify management classes. To modify a management class, go to the Management Class Application Selection panels, shown in Figure 25 on page 68 and Figure 26 on page 69, and specify the name of the SCDS containing the management class you want to change. Specify the management class name and select option 4, Alter. ISMF displays the Management Class Alter panel shown in Figure 96.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                         MANAGEMENT CLASS ALTER              Page 5 of 5
Command ===>

SCDS Name . . . . . . : SMS.ONE.TWO
Management Class Name : MCUSER01

To ALTER Management Class, Specify:
    ALTER Backup Attributes:
    # Versions . . . . . . . .         (1 to 9999, NOLIMIT or blank)
    Retain Only Version  . . .         (1 to 9999, NOLIMIT or blank)
      Unit . . . . . . . . . .         (D=days, W=weeks, M=months, Y=years or
                                       blank)
    Retain Extra Versions  . .         (1 to 9999, NOLIMIT or blank)
      Unit . . . . . . . . . .         (D=days, W=weeks, M=months, Y=years or
                                       blank)
    Copy Serialization . . . .         (C=continue, F=fail or blank)
    Abackup Copy Technique . . S       (P=Conc Preferred, R=Conc Required or
                                       S=Standard)


Use ENTER to Perform Verification; Use UP Command to View previous Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 96. Altering a Management Class

This panel contains the same attributes as the Management Class Define panel. The fields are primed with the information you entered when you defined the management class. To alter the management class, type over the old information with your new information on this panel and on the next three Management Class Alter panels. You can use the DOWN command to reach the next panels and use the END command to save the changes and return to the Management Class Application Selection panel. The changes take effect when you activate this updated configuration.

The # Versions field specifies the number of versions of datasets assigned this management class to be maintained.

The Retain Only Version and Retain Extra Versions fields specify the time period for which the copy is to be retained. The unit of measure for the time period is specified in the accompanying Unit field. This field cannot be blank if the Unit field is specified.

The Copy Serialization field specifies whether aggregate backup should continue if an enqueue failure is encountered.

The Abackup Copy Technique field specifies whether *concurrent copy* should be used during aggregate backup processing.

Valid values are:

P indicates that concurrent copy is preferred and should be used for backup. A data set is backed up on a non-concurrent copy volume if it does not reside on a volume supported by concurrent copy, or if the volume on which it resides is unavailable for concurrent copy.

R indicates that concurrent copy must be used for backup. Backup fails for data sets that don't reside on volumes supported by concurrent copy, or that are unavailable for concurrent copy.

S indicates standard allocation, in which data sets are backed up without using concurrent copy.

The changes that you make take effect on existing data sets at the next scheduled or requested processing time for those data sets.

For information on the results of altering pre-DFSMS/MVS 1.1 Aggregate Groups on a DFSMS/MVS 1.1 system, and vice versa, see “Processing Aggregate Groups” on page 194.

Altering Storage Classes

As the needs of your complex change and as new hardware becomes available, you might need to change the storage classes. To modify a storage class, go to the Storage Class Application Selection panel, shown in Figure 32 on page 85, and specify the name of the SCDS containing the storage class you want to change. Specify the name of the storage class and select option 4, Alter.

ISMF displays the Storage Class Alter panel shown inFigure 97 on page 199.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                              STORAGE CLASS ALTER            Page 1 of 2
Command ===>

SCDS Name . . . . . : USER8.TEST.SCDS
Storage Class Name  : SC1
To ALTER Storage Class, Specify:
  Description ==> TESTING
              ==>
  Performance Objectives
   Direct Millisecond Response . . . .             (1 to 999 or blank)
   Direct Bias . . . . . . . . . . . .             (R, W or blank)
   Sequential Millisecond Response . .             (1 to 999 or blank)
   Sequential Bias . . . . . . . . . .             (R, W or blank)
   Initial Access Response Seconds . .             (0 to 9999 or blank)
   Sustained Data Rate (MB/sec)  . . .             (0 to 999 or blank)
  Availability . . . . . . . . . . . . N           (C, P ,S or N)
  Accessibility  . . . . . . . . . . . N           (C, P ,S or N)
   Backup  . . . . . . . . . . . . . .             (Y, N or Blank)
   Versioning  . . . . . . . . . . . .             (Y, N or Blank)

Use ENTER to Perform Verification; Use DOWN Command to View next Page;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 97. Altering a Storage Class

This panel contains the same attributes as the Storage Class Define panel. The fields are primed with the information you entered when you defined the storage class. To alter the storage class, type over the old information with your new information on this and the next Storage Class Alter panel. You can use the DOWN command to reach the next panel and use the END command to save the changes and return to the Storage Class Application Selection panel. The changes take effect when you activate this updated configuration.

The changes that you make take effect on existing data sets or objects at the next scheduled or requested processing time for those data sets or objects (for example OSMC cycle for objects).

Altering Data Classes

As the needs of your installation change, you might need to change data classes. To modify a data class, go to the Data Class Application Selection panel, shown in Figure 44 on page 116, and specify the name of the SCDS containing the data class you want to change. Specify the data class name and select option 4, Alter.

ISMF displays the Data Class Alter panels shown in Figure 98 on page 200 and Figure 99 on page 200.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                              DATA CLASS ALTER                 Page 1 of 3
Command ===>

SCDS Name  . . : SMS.SCDS1.SCDS
Data Class Name: DATAV

To ALTER Data Class, Specify:
  Description ==> DATACLAS DA000001 FOR VARIABLE LENGTH RECORD
              ==> FORMAT
  Recorg  . . . . . . . .               (KS, ES, RR, LS or blank)
  Recfm . . . . . . . . . VB            (any valid RECFM combination or blank)
  Lrecl . . . . . . . . . 255           (1 to 32761 or blank)
  Keylen  . . . . . . . .               (0 to 255 or blank)
  Keyoff  . . . . . . . .               (0 to 32760 or blank)
  Space Avgrec  . . . . . U             (U, K, M or blank)
        Avg Value . . . . 255           (0 to 65535 or blank)
        Primary . . . . . 5000          (0 to 999999 or blank)
        Secondary . . . . 5000          (0 to 999999 or blank)
        Directory . . . .               (0 to 999999 or blank)

Use ENTER to Perform Verification; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 98. Data Class Alter Panel, Page 1 of 3

Press Enter to see Page 2 of the Data Class Alter panel.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                             DATA CLASS ALTER            Page 2 of 3
Command ===>

SCDS Name . . . :
Data Class Name :
To ALTER Data Class, Specify:
  Retpd or Expdt  . . . . . . __________ (0 to 9999, YYYY/MM/DD or blank)
  Volume Count  . . . . . . . 1          (1 to 59 or blank)
    Add'l Volume Amount . . . _          (P=Primary, S=Secondary or blank)
  Imbed . . . . . . . . . . . _          (Y, N or blank)
  Replicate . . . . . . . . . _          (Y, N or blank)
  CIsize Data . . . . . . . . _____      (1 to 32768 or blank)
  % Freespace CI  . . . . . . ___        (0 to 100 or blank)
              CA  . . . . . . ___        (0 to 100 or blank)
  Shareoptions Xregion  . . . _          (1 to 4 or blank)
               Xsystem  . . . _          (3, 4 or blank)
  Compaction  . . . . . . . . _          (Y, N or blank)
  Media Interchange
    Media Type  . . . . . . . _          (1, 2, 3, 4 or blank)
    Recording Technology  . . ___        (18, 36, 128 or blank)
  Use ENTER to Perform Verification; Use UP/DOWN Command to View other Panels;
  Use HELP Command for Help; Use END Command to Save and Exit; Cancel to Exit.
```

Figure 99. Data Class Alter Panel, Page 2 of 3

Press Enter to see Page 3 of the Data Class Alter panel.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                             DATA CLASS ALTER            Page 3 of 3
Command ===>

SCDS Name . . . : USER2.TEST.ACDS
Data Class Name : ATLDC008

To ALTER Data Class, Specify:
  Data Set Name Type  . . . . . . ___  (EXT, HFS, LIB, PDS or blank)
    If Ext  . . . . . . . . . . . _    (P=Preferred, R=Required or blank)
    Extended Addressability . . . N    (Y or N)
    Record Access Bias  . . . . . _    (S=System, U=User or blank)
  Reuse . . . . . . . . . . . . . N    (Y or N)
  Initial Load  . . . . . . . . . _    (S=Speed, R=Recovery or blank)
  Spanned / Nonspanned  . . . . . _    (S=Spanned, N=Nonspanned or blank)
  BWO . . . . . . . . . . . . . . __   (TC=TYPECICS, TI=TYPEIMS, NO or blank)
  Log . . . . . . . . . . . . . . _    (N=NONE, U=UNDO, A=ALL or blank)
  Logstream Id  . . . . . . . . . __________________________
  Space Constraint Relief . . . . N    (Y or N)
    Reduce Space Up To (%)  . . . __   (0 to 99 or blank)

Use ENTER to Perform Verification; Use UP Command to View previous Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit
```

Figure 100. Data Class Alter Panel, Page 3 of 3

These panels contain the same attributes as the Data Class Define panels. The fields are primed with the information you entered when you defined the data class. To alter the data class, type over the old information with your new information on this and the next panel. You can use the DOWN command to reach the next panel and use the END command to save the changes, and return to the Data Class Application Selection panel. The changes take effect when you activate this updated configuration.

The changes you make apply only to new data sets. Existing data sets are not affected.

| In addition, page 2 of this panel has been modified to allow storage administrators to specify a value for the Additional Volume Amount attribute. Additional Volume Amount is applicable only to extended format VSAM multivolume data sets. Additional Volume Amount is specified when Data Set Name Type = EXT and Volume Count>1. The valid values for this attribute are ’P’ (primary), ’S’ (secondary), and blank (not specified; the system default is primary).

Altering Aggregate Groups

As the needs of your installation change, you might need to change aggregate groups. To modify an Aggregate Group, go to the Aggregate Group Application Selection panel, shown in Figure 48 on page 133, and specify the name of the SCDS containing the aggregate group you want to alter. Specify the aggregate group name and select option 4, Alter.

ISMF displays the Aggregate Group Alter panel shown in Figure 101 on page 202.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                           AGGREGATE GROUP ALTER               Page 1 of 2
Command ===>

SCDS Name  . . . . . : SMS.SCDS1.SCDS
Aggregate Group Name : TESTAG

To DEFINE Aggregate Group, Specify:
 Description ==>
             ==>

 Backup Attributes
   Number of Copies  . . . . . . . 1                (1 to 15)
   Management Class Name . . . . .                  (1 to 8 characters, to be
                                                    defined in current SCDS)

 Output Data Set Name Prefix . . .
                                                    (1 to 33 Characters)
 Account . . . . . . . . . . . . .
                                                    (1 to 32 Characters)

Use ENTER to Perform Verification; Use DOWN Command to View next Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 101. Aggregate Group Alter, Page 1 of 2

The Aggregate Group Alter panels contain the same attributes as the Aggregate Group Define panels discussed in “Defining Aggregate Group Attributes” on page 133. The fields are primed with the information you entered when defining the aggregate group. To alter the aggregate group, simply type over the old information with your new information on this and the next panel. Use the END command to save the changes.

The first page of the Aggregate Group Alter panel contains aggregate group alter attributes. The SCDS Name and Aggregate Group Name fields are output fields that contain the SCDS and aggregate group names you specified on the Aggregate Group Application Selection panel. The Description field is an optional field of 120 characters in which you can describe the aggregate group.

You can specify the following *required* attributes on the first page of the Aggregate Group Alter panel:

Number of Copies
: specifies the number of aggregate backup output files to be altered. The valid values are 1 to 15.

Management Class Name
: specifies the management class name from which the aggregate backup attributes are obtained. The valid values are 1 to 8 alphanumeric characters (first character not a digit) or a blank.

Editing Aggregate Group Attributes

After specifying your backup attribute values, issue the DOWN command to view the second page of the Aggregate Group Alter panel, shown in Figure 102 on page 203.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                           AGGREGATE GROUP ALTER              Page 2 of 2
Command ===>


SCDS Name  . . . . . : SMS.SCDS1.SCDS
Aggregate Group Name : AGUSER03
To Edit a Data Set, Specify Number  . .     (1, 2, 3, 4, 5, or 6)
  Selection Data Sets:            (1 to 44 characters)
   1 ===>
     Member Name ===>             (1 to 8 characters)
   2 ===>
     Member Name ===>             (1 to 8 characters)
   3 ===>
     Member Name ===>             (1 to 8 characters)
   4 ===>
     Member Name ===>             (1 to 8 characters)
   5 ===>
     Member Name ===>             (1 to 8 characters)

  Instruction Data Set:           (1 to 44 characters)
   6 ===>
Use ENTER to Perform Verification; Use UP Command to View previous Panel;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 102. Aggregate Group Alter, Page 2 of 2

The second page of the Aggregate Group Define panel contains the selection and instruction data set names for the aggregate group. The SCDS Name and Aggregate Group Name fields are output fields that contain the SCDS and aggregate group names you specified on the Aggregate Group Application Selection panel.

Edit a Data Set
: Select the number of a selection or instruction data set that you want to edit. When you select a data set number, it lets you alter the data set by invoking PDF Edit. The PDF edit screen is shown in Figure 51 on page 136. See *OS/390 ISPF User's Guide* for more information on the PDF Edit commands.

Selection Data Set Name
: Name of the data set containing lists of data sets to be included in the application backup. You can specify up to five selection data set names. One data set name is required. There is no default. If you want to enter a fully qualified data set name, enclose the name in single quotes. If you do not enclose the name in single quotes, the TSO prefix is added to the name as the first high level qualifier. There is no default.

Member Name
: Name of the data set member containing lists of data sets to be included in the application backup if the selection data set is partitioned. This name must be a valid TSO data set member name. If the data set specified in the Selection Data Set Name field is a partitioned data set, you must specify a valid member name. There is no default.

Instruction Data Set Name
: Name of the data set containing instruction, commands, etc., that are copied into the control file volume after the backup control file. This data set can only be a sequential data set. You must use a valid TSO data set name. The data set name, including the TSO prefix, can be no more than 44 characters long. This is an optional field and has no default.

The changes take effect when you activate this updated configuration.

For information on the results of altering pre-DFSMS/MVS 1.1 Aggregate Group constructs on a DFSMS/MVS 1.1 system, and vice versa, see “Processing Management Classes” on page 195.

Copying SMS Classes, Storage Groups, and Aggregate Groups

To simplify the creation of new SMS classes, storage groups, aggregate groups, and SCDSs, you can copy existing ones and modify them. In the LINE OPERATOR column of any List panel, you can enter the COPY line operator and press ENTER to duplicate an existing SMS class, storage group or aggregate group. You receive the Copy Entry panel shown in Figure 103.

You should generate your list from an SCDS before issuing a COPY command. Note also that for storage groups, the volume list associated with the storage group is not copied. See *DFSMS/MVS OAM Planning, Installation, and Storage Administration Guide for Object Support* for more information on copying optical libraries and drives.

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                               COPY ENTRY PANEL
Command ===>

Construct will be Copied from:

  Data Set Name . . : SMS.SCDS1.SCDS
  Construct Name  . : DATAV
  Construct Type  . : MANAGEMENT CLASS

Specify "Copy To" Construct:

  Data Set Name . . . 'SMS.SCDS2.SCDS'
                                (1 to 46 Characters)

  Construct Name  . .           (1 to 8 characters, fully specified)

Enter "/" to select option   _  Replace like-named Construct

                             _  Perform Alter

Use ENTER to Perform COPY;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 103. Copy Entry Panel

The *from* Data Set Name field identifies the source you are copying. It is primed with the value you specified on the Application Selection panel. The *from* Construct Name field identifies the name of the SMS class or storage group you are copying. It is primed with the value from the Name field of the List panel. The *from* Construct Type field identifies whether you are copying a data class, storage class, management class, aggregate group or storage group. It is primed with the type of List panel from which you issued the COPY command. You should generate your list from an SCDS instead of the ACDS prior to copying SMS classes, storage groups, and aggregate groups.

The *to* Data Set Name field identifies the target of the copy. It must be the name of an SCDS. It is primed with the value of the *from* Data Set Name if the *from* Data Set Name contains an SCDS name. The *to* Construct Name field identifies the name of the SMS class or storage group copy. It is primed with blanks. The copy has the same *type* as the original.

If the SMS class, storage group, or aggregate group that you identify in the *to* fields already exists, you specify whether you want to replace it in the Replace like-named Construct field. If you indicate Y, yes, the replacement occurs. If you indicate N, no, the copy does not occur. The field is primed with N, which is the default.

In the Perform Alter field, you indicate if you want to change some of the attributes of the copy you are creating. If you specify Y, yes, the system displays the Management Class Application Selection panel. This panel lists four groups of management class attributes. You can then choose an attribute group for altering ahead of other attributes. If you specify N, no, you remain on the Copy Entry panel, where you can perform another copy or return to the original List panel. The field is primed with N, which is the default.

When you have specified the values, press ENTER to perform the copy.

Note: The individual constructs cannot be copied to an SCDS of a pre-DFSMS/MVS 1.1 system. The individual fields within the constructs are larger on the higher level system because of new functions.

Deleting DASD Volumes from the System

You might want to delete DASD volumes from the system. You can do this without performing an initial program load (IPL) by following this procedure:

1. Set the DASD volume status to DISNEW. This disables any new allocations to the volume.
2. Move data sets that you want to keep to other volumes using DFSMSdss.
3. Delete the DASD volume from the storage group that contains it. See "Deleting DASD Volumes from Storage Groups" for information on deleting DASD volumes.
4. Vary the DASD volume offline. If you need specific information about how to vary a volume offline, refer to *DFSMS/MVS DFSMShsm Storage Administration Guide* for guidance.
5. Issue the DFSMShsm DELVOL command from all DFSMShsm systems that are aware of the volume. This prevents DFSMShsm from attempting to allocate to the volume during subsequent interval migrations. If you need specific information on the DFSMShsm DELVOL command, refer to *DFSMS/MVS DFSMShsm Storage Administration Guide* for guidance.

Deleting DASD Volumes from Storage Groups

When you delete a DASD volume from a storage group, first set the volume status to DISNEW. This prevents any new allocations to the volume while allowing existing data sets to still be accessed with DISP=SHR, DISP=MOD, or DISP=OLD. Move any data that you do not want converted to non-SMS from the volume. Next, if the DASD volume is to be reused as a non-SMS-managed volume, run a DFSMSdss NONSMS CONVERTV to convert it out of SMS. Finally, remove the volume from the storage group.

To remove DASD volumes from a storage group, select the Storage Group Application Selection panel. Specify the CDS name and the storage group name that contains the volumes you want to delete. Press ENTER to get the Storage Group Volume Selection panel shown in Figure 95 on page 196. From the Storage Group Volume Selection panel, indicate the volumes you want to delete in the SPECIFY A SINGLE VOLUME (in PREFIX), OR RANGE OF VOLUMES field, select option 4, and press ENTER. Be careful when moving or removing a volume from a storage group because the volume could contain part of a multivolume data set. The changes take effect when you activate this updated configuration.

To prevent DFSMShsm from attempting to allocate to volumes which have been deleted from the storage group, issue the DFSMShsm DELVOL command from all DFSMShsm systems which are aware of the deleted volume.

After deletion, the DASD volume is only eligible for non-SMS allocations. However, if you are reassigning the volume to another storage group, you need to define the volume to the storage group and then activate the updated configuration.

Deleting Storage Groups

You might want to delete a pool type storage group so that you can define all of its volumes to other storage groups or so that you can use the volumes for non-SMS allocations. The following sections describe the procedure for deleting storage groups and some considerations for preventing job failures.

Deleting or Moving System-Managed Data Sets from the Storage Group

If you plan to use the volumes for non-SMS allocations, you need to delete or move all system-managed data sets. The following procedure shows how to do this and how to remove the storage group from the SMS configuration:

1. Set the storage group status to DISNEW to prevent any further new allocations to volumes defined to the storage group. As time passes, DFSMShsm space management processing clears most of the data sets off the volumes in the storage group you are deleting, provided AUTO MIGRATE is ‘Y’ and most of the data sets have an associated management class.
2. Use DFSMSdss to move any remaining data sets that have lingered on the storage group volumes when the volumes are sufficiently empty.
3. Build an SCDS to contain the new configuration. Remove the storage group definition from the SCDS and change the storage group ACS routine so that it does not select the deleted storage group.
4. Activate the new configuration.

To declare volumes non-SMS-managed, you need to INIT them through ICKDSF to non-SMS or CONVERT them through DFSMSdss to non-SMS.

When you delete a VIO or dummy type storage group, you do not have to worry about redefining volumes to other storage groups, but you might need to modify the storage group ACS routine.

Preventing Job Failures

You should be aware that if you delete a storage group while a job that uses this group is running, the job might fail with MSGIGD17201I. To prevent this, consider performing the following:

1. Modify your storage group ACS routine so that it does not assign the storage group to be deleted to any new allocations. Validating your configuration causes MSGIGD06023I to be received. This message is only a warning. As long as there are no error messages, your configuration is valid.
2. Activate the configuration with the modified storage group ACS routine and allow it to remain active until jobs that used the old ACS routine have completed. This allows executing jobs to complete using the existing storage group definition.
3. Delete the storage group from the configuration. Since the storage group no longer exists in the configuration, you do not receive MSGIGD06023I.
4. Activate the configuration from which the storage group has been deleted.

Deleting SMS Classes

You need to exercise caution when you delete storage, management and data classes, because a user or system might attempt to use a data set that references them.

For example, assume a data set has some specified migration attributes, and at some point it meets the migration criteria. After the data set has been migrated, you delete its associated management class. If the data set is recalled in the future, it must go through the management class ACS routine. If the ACS routine fails to override the undefined or deleted management class, the recall fails because a management class that does not exist is associated with the data set. If this occurs, you must either rewrite the management class ACS routine or specify that the data set should be recalled as a non-SMS data set. If you rewrite the ACS routine, you must check for the deleted management class and assign a valid management class to the data set. If you decide to bypass ACS processing, be aware that the data set is recalled to non-SMS storage.

Another consequence of deleting management classes involves DFSMShsm automatic processing. When DFSMShsm runs through its automatic cycle and processes data sets on the basis of their management class attributes, it attempts to retrieve nonexistent management class definitions. Consequently, DFSMShsm skips the processing of these data sets.

Instead of deleting storage classes and management classes, you should prevent any new allocation from using them by rewriting the corresponding ACS routine to override the deleted storage class or management class. It is safer and provides the possibility that all data sets that reference the management class or storage class might eventually be overridden. Also, you can identify all data sets with storage class or management class and then use ALTER to change to a new storage class or management class. If you decide to delete a storage class or management class, make certain that you inform all users well in advance.

Because data classes are used only at data set allocation, they do not have these problems. The original data class name is never referenced or reused.

If you decide to delete an SMS class, enter the following information on the pertinent Class Application Selection panel:

```
SCDS NAME         ===> 'SMS.SCDS1.SCDS'
xxxx CLASS NAME   ===> *
SELECT OPTION     ===> 1
```

After you press ENTER, ISMF displays the Class List panel. Next to the SMS class that you want to delete, enter DELETE in the LINE OPERATOR column. When you press ENTER, you see the Confirm Delete Request panel shown in Figure 104. Confirm that the displayed SMS class is the one that you want to delete. If it is,

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                            CONFIRM DELETE REQUEST
Command ===>

  To Confirm Deletion on the following element:

     Element Name . . : SYS4
     Element Type . . : SYS GROUP NAME
     Residing in SCDS : SMS.SCDS1.SCDS


  Specify the following:
      Enter "/" to select option   /  Perform Deletion







Use ENTER to Perform Operation;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 104. Confirm Delete Request Panel

enter Y for yes and press ENTER. The SMS Class List should appear with '*DELETE' in the LINE OPERATOR column next to the deleted SMS class.

After deleting SMS classes, activate the modified SMS configuration to make your changes part of the active configuration.

Converting Volumes to SMS

You can convert empty volumes through ISMF using the INIT line operator from the Volume application. You can convert previously-used, non-SMS-managed volumes using the CONVERTV command of DFSMSdss. You can also unconvert SMS volumes using the CONVERTV command of DFSMSdss.

See *DFSMS/MVS Implementing System-Managed Storage* for an explanation of how to convert and unconvert volumes.

Chapter 12. Recovering Storage Management Subsystem Information

This chapter describes how to recover SMS information.

Recovering Control Data Sets

You should allocate a spare ACDS and COMMDS when you allocate your originals. Having spare copies eases the recovery process if SMS cannot access the originals because of I/O errors. Place the spares on a device that every system in your complex can access. Make certain that they reside on a different device from your originals.

Remember that in order to support 32 names, an SMS control data set is converted from 8-name mode to 32-name mode. You must confirm this conversion, using the operator console or via ISMF.

Note: There are reasons for making copies of your control data sets before the system mode is converted:

- The conversion is permanent.
- You might want to maintain them for compatibility with down-level systems.

| Because you cannot define or alter an ACDS, you cannot use it as an SCDS
| if the SCDS is lost.

Recovering an SCDS

For purposes of recovery, treat an SCDS the same as any other source VSAM linear data set. You can use DFSMShsm to manage its availability, which relieves you from having to allocate any spares.

Recovering an ACDS

You can recover from errors that prevent access to the ACDS if you have allocated a spare. All permanent errors that make the ACDS unreadable or unwritable require intervention. For permanent I/O errors to the ACDS, the messages IGD041I and IGD040D appear on the operator console. You have the option of re-trying the failing operation or using the spare ACDS.

If you decide to use the spare ACDS, you need to perform three steps.

1. Reply 'C' on the operator console.
2. Copy the data from the SMS address space to the spare ACDS by issuing the following command from a system in the SMS complex:
   ```
   SETSMS SAVEACDS(spare.acds)
   ```
3. Tell the system to use the spare ACDS by issuing the following command:
   ```
   SETSMS ACDS(spare.acds)
   ```

You need only to issue the `SETSMS ACDS(spare.acds)` command on one system. The COMMDS is updated to reflect this change. As the other systems in the SMS complex access the COMMDS (based on the INTERVAL value in their respective IGDSMSxx members), they automatically switch to the new ACDS.

Recovering a COMMDS

If you cannot access the COMMDS, you can recover from the error if you allocated a spare data set. All permanent errors that make the COMMDS unreadable require intervention. For permanent I/O errors to the COMMDS, the messages IGD041I and IGD070D appear on an operator console. Reply 'S' on the operator console and issue the following command from a system in the SMS complex:

```
SETSMS COMMDS(spare.commds)
```

One of three situations results.

1. If the spare COMMDS is empty it gets formatted automatically, and SMS writes the in-storage copy of the current COMMDS into the `spare.commds`. You then need to issue the following command on each of the remaining systems in the SMS complex:
   ```
   SETSMS COMMDS(spare.commds)
   ```
2. If the spare COMMDS is not empty but describes an ACDS that is not currently active in the SMS complex, then SMS issues the message IGD076D. This message asks if you want to use the contents of the COMMDS and the ACDS to which it points. Reply 'C' to cause SMS to replace the contents of the `spare.commds` with the in-storage copy of the current COMMDS. You then need to issue the following command on each of the remaining systems in the SMS complex:
   ```
   SETSMS COMMDS(spare.commds)
   ```
3. If the spare COMMDS is not empty but describes the ACDS that is currently active in the SMS complex, you need to issue the following command on each of the remaining systems in the SMS complex:
   ```
   SETSMS COMMDS(spare.commds)
   ```

A response of ’S’ to IGD070D is recommended when recovering from the current COMMDS because a response of ’C’ might result in an unrecoverable error when trying to re-access the current COMMDS. When access to the current COMMDS is suspended, SMS is able to access the new COMMDS without accessing the current COMMDS and resulting in further errors.

Without a usable COMMDS, the systems in the SMS complex have no means of communication. Other systems in the SMS complex are aware of the error, but they are unaware of the switch to a new COMMDS until you inform them.

If you can access the current COMMDS but you want to use an alternate one, you only need to issue the `SETSMS` command from one system. The other systems in the SMS complex detect the change from the old COMMDS to the new COMMDS and they automatically switch to the new one.

Notes on Recovering Control Data Sets

If an SMS system goes down (a temporary condition) or has not yet been started with SMS, then it is not part of the system-managed storage environment. When you activate SMS on the system, it uses the ACDS and COMMDS that are specified in its IGDSMSxx member. If the COMMDS and ACDS are different from the ones used by the remainder of the SMS complex, then the system runs as a separate SMS complex.

Consequently, when you change the current ACDS or COMMDS, update the IGDSMSxx member of SYS1.PARMLIB for each system in the SMS complex.

There are special considerations to keep in mind when you move control data sets and their catalog entries. For more information on this, including the use of REPRO MERGECAT, see *DFSMS/MVS Managing Catalogs*.

Recovering from a Systems Failure in the SMS Complex

To provide for recovery procedures, you should leave one system or system group name slots empty in the corresponding function panels. In case a problem occurs requiring that a system that is defined within the configuration as part of a system group be specified by its specific name (i.e., hardware failure), you can add it without disrupting the rest of the system group of SMS complex.

You can initiate recovery with the following steps:

- Use the ISMF CDS Application to add the system name to the SMS configuration.
- Activate the configuration, either from ISMF or from the operator console.
- Use the VARY SMS operator command to make any necessary changes to the configuration.
- Remove the system name from the configuration and reactivate the configuration once the problem has been corrected.

Recovering the SMS Address Space

If the SMS address space fails, SMS automatically attempts to restart up to six times. If SMS fails to recover its address space after the sixth restart, you have two options:

1. You can end the SMS address space and then restart SMS using the `T SMS=xx` command. If you have cross memory interactions, users' address spaces wait for the SMS address space to restart. After the sixth automatic restart, SMS displays a message requesting the user to select the next action; retry another restart or terminate the SMS address space. This relates to the SMS address space and *not* the user's address space.
2. You can allow SMS to attempt another restart by replying "YES" to the system-generated message:

   ```
   IGD032D
   ```

Canceling the SMS Address Space

You cannot cancel SMS once it is activated. You must re-IPL to cancel the SMS address space and deactivate SMS. If the system is set up with automatic activation of SMS at IPL time and you want to deactivate SMS, you need to modify the appropriate SYS1.PARMLIB members to prevent automatic activation of SMS.

Chapter 13. Protecting the Storage Management Subsystem

To ensure that SMS operates correctly, you need to prevent unauthorized end users from modifying information in certain control data sets, such as SMS class, aggregate group and storage group definitions. Some ISMF commands and functions also require protection from use by unauthorized users. This chapter explains how to use RACF (Resource Access Control Facility) to establish authorization levels for protecting these data sets, commands, and functions.

Data set password protection does not apply to data sets or catalogs managed by SMS. SMS assumes the presence of RACF or an equivalent security product. Password protection is bypassed for all system-managed data sets.

Identifying the Resource Owner and Extracting the Default Classes

The *resource owner* is the actual owner of the data set covered by the RACF DATASET profile. RACF extracts the resource owner based on data set name. If the resource owner is not identified, the high-level qualifier is used.

If you specify ACSDEFAULTS=YES in the IGDSMSxx member, RACF uses the resource owner to extract the default SMS classes and application identifier. If the resource owner is a user and no default SMS information is available from the user profile, the default information from the group profile is used. If the resource owner is a group, then the defaults for the group profile is used. You can protect the ability to update the resource owner field, RESOWNER, in the data set RACF profile. Revoked USERID should not be used as a resource owner, or it causes RACF to fail. See also “FIELD Resource Class” on page 226.

If you specify USE_RESOWNER=NO in the IGDSMSxx member, RACF uses the execution user ID instead of the resource owner to check authorization. This allows users who do not use a naming convention, user ID or group ID as the high level qualifier of data set names to check authorization to use storage and management classes. If you specify USE_RESOWNER=YES, there is no change to current processing. This is the default.

After ACS routines have been run, RACF is invoked to verify the user’s authority to allocate the data set (CREATE/ALTER) and the resource owner’s authority to use the STORCLAS and MGMTCLAS (READ). You can protect the ability of a resource owner to use management class and storage class through STORCLAS and MGMTCLAS resource classes.

See *MVS/ESA SML: Managing Data* for additional information on using RACF in an SMS environment.

Protecting ISMF Functions

You can use RACF authorization to limit access to the following categories of ISMF functions:

1. The entire ISMF component
2. The individual ISMF applications:

   Profile
      Data Set

DASD Volume
Mountable Optical Volume

Management Class
Data Class

Storage Class
Storage Group

Automatic Class Selection
Control Data Set

Aggregate Group
Library Configuration

Library Configuration
Drive Configuration

Data Collection
List

3. The ISMF line operators
4. The ISMF commands

ISMF relies on the *RACF program control feature* to protect many of its applications. The RACF program control feature prevents unauthorized end users from running selected ISMF programs. To use the feature, you must activate the RACF Program Class and define your selected ISMF programs to RACF.

With RACF program control you can set up authorization levels for each of these categories, varying the level within a particular category to suit the needs of your installation. Individual end users can execute an ISMF function if one of the following conditions is true:

- They are authorized to execute the corresponding load module.
- Their RACF profile contains the OPERATIONS attribute.
- Their group is authorized to execute the load module.
- RACF is disabled or the program control feature is turned off.
- The universal access authority (UACC) for the load module is READ or greater, making the load module available to anyone who can access ISMF.

Note: In order to make sure that only particular users can use the storage administrator applications and functions, these functions should be protected with RACF program control, and not just by user mode level. This is because a TSO user could change his user mode level, as this information is contained in the user's ISPF profile.

The RACF program resource class allows the security administrator to protect various ISMF applications and functions with program control. This is achieved by controlling the access to load modules which are invoked by:

- ISMF Applications
- ISMF Line Operators
- ISMF Commands

The load modules reside in the following libraries:

- SYS1.DGTLLIB for DFSMSdfp/ISMF
- SYS1.DGTLLIB for DFSMSdss/ISMF
- SYS1.DFQLLIB for DFSMShsm

If the installation moves these modules to another load library, the installation-defined load library must be used in the program protection.

To protect a load module, use the RDEFINE RACF command. The syntax of this command is:

```
RDEFINE PROGRAM mod-name OWNER(owner of profile)         +
        UACC(NONE) ADDMEM('dsn of loadlib'/volser/NOPADCHK)
```

See *OS/390 Security Server (RACF) Security Administrator's Guide* for a detailed description of how to use the RACF program control features.

Locating Module Names for ISMF Applications

NOT Programming Interface information

Table 10 lists the load module names you can use to limit access to certain ISMF applications. The names are included in the panel coding for the ISMF Primary Option Menu for Storage Administrators, which is stored in the panel library with a member name of DGTSMMD2. If you want to limit access to all of ISMF, use module name DGTFMD01.

Table 10. Module Names for ISMF Applications

Application	**Module Name**
Profile	DGTFPF00
Data set	DGTFDS00
DASD Volume	DGTFVA00
Mountable Optical Volume	DGTFOVCD
Management class	DGTFMCCD
Data class	DGTFDCCD
Storage class	DGTFSCCD
Storage group	DGTFSGDR
Automatic class selection	DGTFFLAD
Control data set	DGTFSACD
Aggregate Group	DGTFAGCD
Optical Library Configuration	DGTFLCCD
Optical Drive Configuration	DGTFRCCD
Data Collection	DGTFADAD
List	DGTFJLCD
Mountable Tape Volume	DGTFTVCD
Tape Library Configuration	DGTFLMCD

End of NOT Programming Interface information

Locating Module Names for ISMF Functions

Restricting access to the DGTFPF05 module prevents end users from gaining access to the Primary Option Menu for Storage Administrators. Table 11 on page 216 lists this and the other load module names you can use to limit access to certain ISMF functions.

Table 11. Module Names for ISMF Functions

Function	Module Name
User mode	DGTFPF05
Logging and abend control	DGTFPF02
ISMF job statement information	DGTFPF03
DFSMSdss execute statement information	DGTFPF04
ICKDSF execute statement information	DGTFPF20
Data set print execute statement information	DGTFPF21
IDCAMS execute statement information	DGTFPF22
Data Class DEFINE	DGTFDCDA
Data Class ALTER	DGTFDCAA
Data Class DISPLAY	DGTFDCDI
Data Class LIST	DGTFDCLD
Storage Class DEFINE	DGTFSCDA
Storage Class ALTER	DGTFSCAA
Storage Class DISPLAY	DGTFSCDI
Storage Class LIST	DGTFSCLD
Management Class DEFINE	DGTFMCDA
Management Class ALTER	DGTFMCAA
Management Class DISPLAY	DGTFMCDI
Management Class LIST	DGTFMCLD
Storage Group DEFINE	DGTFSGFR
Storage Group ALTER	DGTFSGAR
Storage Group LIST	DGTFSGLD
Storage Group VOLUME	DGTFSGVR
Aggregate Group DEFINE	DGTFAGDA
Aggregate Group ALTER	DGTFAGAA
Aggregate Group DIQPLAY	DGTFAGDI
Aggregate Group LIST	DGTFAGLD
Optical Library Configuration DEFINE	DGTFLCDE
Optical Library Configuration ALTER	DGTFLCAL
Optical Library Configuration DISPLAY	DGTFLCDI
Optical Library Configuration LIST	DGTFLCLD
Optical Drive Configuration DEFINE	DGTFRCDE
Optical Drive Configuration ALTER	DGTFRCAL
Optical Drive Configuration DISPLAY	DGTFRCDI
Optical Drive Configuration LIST	DGTFRCLD
Tape Library Configuration DEFINE	DGTFLMDE
Tape Library Configuration ALTER	DGTFLMAL
Tape Library Configuration DISPLAY	DGTFLMDI
Tape Library Configuration LIST	DGTFLMLD

Locating Module Names for Line Operators and Commands

NOT Programming Interface information

Table 12, Table 13 on page 218, Table 14 on page 219 and Table 15 on page 219 list the module names for ISMF line operators and commands. The names are stored in command tables in the DGTLLIB load library. The line operators for storage administrators are described in "Appendix A. ISMF Command and Line Operator Reference Summary" on page 287. Line operators for end users are described in *DFSMS/MVS Using ISMF*.

Note: You can invoke CATLIST or VTOCLIST from the Data Set List or outside of ISMF, but they are not ISMF line operators with ISMF load module names.

End of NOT Programming Interface information

NOT Programming Interface information

Table 12. Module/CLIST Names for ISMF Line Operators, Part I

Line Operator	Aggregate Group List	Data Class	Data Set List	Optical Drive List
ALTER	DGTFALH1	DGTFALD1	DGTFAL01	DGTFALR1
ANALYZE	-	-	-	-
BROWSE	-	-	DGTFBR01	-
CLIST	-	-	DGTFCL01	-
COMPRESS	-	-	DGTFCM01	-
CONDENSE	-	-	DFQFCND1	-
CONVERTV	-	-	-	-
COPY	DGTFCAH1	DGTFCAD1	DGTFCY01	DGTFCAR1
DEFRAG	-	-	-	-
DELETE	DGTFDNH1	DGTFDND1	DGTFDL01	DGTFDNR1
DISPLAY	DGTFDIH1	DGTFDID1	-	DGTFDIR1
DUMP	-	-	DGTFDP01	-
EDIT	-	-	DGTFED01	-
EJECT	-	-	-	-
ERASE	DGTFDNH1	DGTFDND1	DGTFDL01	DGTFDNR1
HALTERDS	-	-	DFQFHA01	-
HBACKDS	-	-	DFQFHB01	-
HBDELETE	-	-	DFQFHBD1	-
HDELETE	-	-	DFQFHD01	-
HIDE	DGTFHI01	DGTFHI01	DGTFHI01	DGTFHI01
HMIGRATE	-	-	DFQFHM01	-
HRECALL	-	-	DFQFHRL1	-
HRECOVER	-	-	DFQFHRC1	-
INIT	-	-	-	-
INSPECT	-	-	-	-
LIST	-	-	-	-
LISTSYS	-	-	-	-
LISTVOL	-	-	-	-
MESSAGE	DGTFMS00	DGTFMS00	DGTFMS00	DGTFMS00
RAUTH	-	-	-	-
RECOVER	-	-	-	-
RELEASE	-	-	DGTFRL01	-
RESTORE	-	-	DGTFRT01	-
REFORMAT	-	-	-	-

Table 12. Module/CLIST Names for ISMF Line Operators, Part I (continued)

Line Operator	Aggregate Group List	Data Class	Data Set List	Optical Drive List
SECURITY	DGTFSRD1	-	DGTFSRD1	-
SETCACHE	-	-	-	-
STATUS	-	-	-	-
TSO Commands and CLIST	DGTFUS01	DGTFUS01	DGTFUS01	DGTFUS01

Table 13. Module/CLIST Names for ISMF Line Operators, Part 2

Line Operator	Optical Library List	List	Management Class	Mountable Optical Volume List
ALTER	DGTFALL1	-	DGTFALM1	-
AUDIT	DGTFAUL1	-	-	DGTFAU01
ANALYZE	-	-	-	-
BROWSE	-	-	-	-
CLIST	-	-	-	DGTFCL01
COMPRESS	-	-	-	-
CONDENSE	-	-	-	-
CONVERTV	-	-	-	-
COPY	DGTFCAL1	-	DGTFCAM1	-
DEFRAG	-	-	-	-
DELETE	DGTFDNL1	DGTFEL01	DGTFDNM1	-
DISPLAY	DGTFDIL1	-	DGTFDIM1	-
DUMP	-	-	-	-
EDIT	-	-	-	-
EJECT	-	-	-	DGTFEF01
ERASE	DGTFDNL1	DGTFEL01	DGTFDNM1	-
HALTERDS	-	-	-	-
HBACKDS	-	-	-	-
HBDELETE	-	-	-	-
HDELETE	-	-	-	-
HIDE	DGTFHI01	DGTFHI01	DGTFHI01	DGTFHI01
HMIGRATE	-	-	-	-
HRECALL	-	-	-	-
HRECOVER	-	-	-	-
INIT	-	-	-	-
INSPECT	-	-	-	-
LIST	-	DGTFLL01	-	-
LISTSYS	-	-	-	-
LISTVOL	DGTFLVL1	-	-	-
MESSAGE	DGTFMS00	DGTFMS00	DGTFMS00	DGTFMS00
RAUTH	-	-	-	-
RECOVER	-	-	-	DGTFRC01
RELEASE	-	-	-	-
REMAP	DGTFRML1	-	-	-
RESTORE	-	-	-	-
REFORMAT	-	-	-	-
SECURITY	-	-	DGTFSRD1	-
SETCACHE	-	-	-	-
STATUS	-	-	-	-
TSO Commands and CLIST	DGTFUS01	DGTFUS01	DGTFUS01	DGTFUS01

Table 14. Module/CLIST Names for ISMF Line Operators, Part 3

Line Operator	Storage Class	Storage Group	Volume	Tape Library	Mountable Tape Volume
ALTER	DGTFALS1	DGTFALG1	-	DGTFALY1	DGTFAL11
ANALYZE	-	DGTFAZ01	DGTFAZ01	-	-
AUDIT	-	-	-	DGTFAUL1	DGTFAU04
BROWSE	-	-	-	-	-
BUILDX	-	-	DGTFBX01	-	-
CLIST	-	-	DGTFCL01	DGTFCL01	DGTFCL01
COMPRESS	-	-	DGTFCS01	-	-
CONDENSE	-	-	-	-	-
CONTROL	-	-	DGTFCT01	-	-
CONVERTV	-	-	DGTFCN01	-	-
COPY	DGTFCAS1	DGTFCAG1	DGTFCV01	DGTFCAY1	-
DEFRAG	-	-	DGTFDF01	-	-
DELETE	DGTFDNS1	DGTFDNG1	-	DGTFDNY1	-
DISPLAY	DGTFDIS1	-	-	DGTFDIY1	-
DUMP	-	-	DGTFDM01	-	-
EDIT	-	-	-	-	-
EJECT	-	-	-	-	DGTFEJ01
ERASE	DGTFDNS1	DGTFDNG1	-	-	-
HALTERDS	-	-	-	-	-
HBACKDS	-	-	-	-	-
HBDELETE	-	-	-	-	-
HDELETE	-	-	-	-	-
HIDE	DGTFHI01	DGTFHI01	DGTFHI01	DGTFHI01	DGTFHI01
HMIGRATE	-	-	-	-	-
HRECALL	-	-	-	-	-
HRECOVER	-	-	-	-	-
INIT	-	-	DGTFIN01	-	-
INSPECT	-	-	DGTFIV01	-	-
INSTALL	-	-	DGTFIL01	-	-
LIST	-	-	DGTFIV01	-	-
LISTSYS	-	DGTFLIC1	-	-	-
LISTVOL	-	DGTFLVC1	-	DGTFLVL1	-
MESSAGE	DGTFMS00	DGTFMS00	DGTFMS00	DGTFMS00	DGTFMS00
RAUTH	-	-	DGTFRA01	-	-
RECOVER	-	-	-	-	-
REFORMAT	-	-	DGTFRF01	-	-
RELEASE	-	-	DGTFRV01	-	-
RESTORE	-	-	DGTFRO01	-	-
REVAL	-	-	DGTFRB01	-	-
SECURITY	DGTFSRD1	-	-	DGTFSRD1	-
SETCACHE	-	-	DGTFCB01	-	-
STATUS	-	-	-	-	
TSO Commands and CLIST	DGTFUS01	DGTFUS01	DGTFUS01	-	-

Table 15. Module/CLIST Names for ISMF Commands

Command	Module Name	CLIST
ACTIVATE	DGTFACAT	-
AUDIT	DGTFAU02	-
BOTTOM	DGTFDO01	-
CLEAR	DGTFCR01	-
COMPRESS	DGTFCP01	-
COPY	DGTFCO01	-

Table 15. Module/CLIST Names for ISMF Commands (continued)

Command	Module Name	CLIST
DOWN	DGTFDO01	-
DSUTIL	-	DSUTIL
DUMP	DGTFDU01	-
ERTB	DGTFER02	-
FILTER	DGTFFI01	-
FILTER CLEAR	DGTFFI01	-
FIND	DGTFFN01	-
FOLD	DGTFFU01	-
LEFT	DGTFLE01	-
LIBRARY	-	LIBRARY
LISTPRT	DGTFPR01	-
MIGRATE	-	MIGRATE
PROFILE	DGTFPF01	-
QRETRIEV (DS)	ACBUTO3	-
QRETRIEV (DVOL)	ACBUTO4	-
QSAVE (DS)	ACBUTO6	-
QSAVE (DVOL)	ACBUTO7	-
RECALL	-	RECALL
RECOVER	DGTFRC01	-
RELEASE	DGTFRE01	-
RESHOW	DGTFRW01	-
RESTORE	DGTFRR00	-
RIGHT	DGTFRI01	-
SAVE	DGTFSLDS	-
SORT	DGTFSO01	-
TOP	DGTFUP01	-
UP	DGTFUP01	-
VALIDATE	DGTFVLVA	-
VIEW	DGTFVW01	-

End of NOT Programming Interface information

If you want to look at the command tables, you need the data set name that your installation uses for the load library. The install process for ISMF puts the load modules in SYS1.DGTLLIB (for DFSMSdfp/ISMF and DFSMSdss/ISMF), and SYS1.DFQLLIB (for DFSMShsm/ISMF). However, your installation's post-install procedures might involve moving the ISMF modules to other libraries. If the libraries have moved but have kept the same names, you can determine library data sets names by issuing the TSO/E LISTALC command and scanning the low-level qualifiers for DGTLLIB and DFQLLIB.

Protecting Modules

Take the following three program control steps to protect modules:

1. Use the RDEFINE command or the RACF ISPF entry panels to identify the modules you want to protect. To define the modules to RACF, supply the name of the load module that you want to protect, the name of the data set that contains the load module, and the volume serial number of the volume that contains the data set. RACF adds each module that you identify to the profile for the PROGRAM general resource class.

   When you define the modules, you have several options:

- If you want to define several modules at the same time, you can use asterisk notation. For example, DGT* represents all of the modules beginning with the letters DGT.
- You can add an access list with user IDs or group names and their associated access authority to the profile.
- You can define the UACC to give default access to all users or to none.
- You can use the AUDIT parameter to set up or to bypass RACF logging.

2. Use the PERMIT command to allow end users to execute an application, line operator, or command associated with a module.
3. To prevent unauthorized users from copying a program, renaming it to a name that is unknown to the program control, and then executing the renamed program, you should protect the PDS libraries containing RACF-controlled programs with a UACC of NONE. In order for users to execute programs in these libraries, place the libraries in the LNKLIST concatenation. See *OS/390 Security Server (RACF) Security Administrator's Guide* for more information.

Storage Administration (STGADMIN) Profiles in the FACILITY Class

In order to control the ability to perform functions associated with storage management, define profiles in the FACILITY class whose profile names begin with STGADMIN (storage administration). Although some of these FACILITY profiles are used to protect the Storage Management Subsystem, they are checked whether or not you are using SMS.

There are two sets of facility class profiles. The first are command and keyword related and the second are oriented towards actions and provide authority for storage administrators to do things like backup or copy data sets to which the administrator would usually have no authority.

Command and Keyword Related Profiles

The RACF FACILITY resource class can control the ability to:

- Activate a configuration
- Perform certain catalog functions on data sets using access method services
- Perform certain DFSMSdss functions on data sets

If defined, these profiles are checked before a user is allowed to perform the protected function. If these profiles are not defined, other RACF or password checking is still made to verify authority. [2] Also, the user program must be Authorized Program Facility (APF)-authorized.

Some FACILITY profiles are not checked if the caller is using the system key or is running in supervisor state. These profiles are:

STGADMIN.IGG.DEFNVSAM.NOBCS
STGADMIN.IGG.DEFNVSAM.NONVR
STGADMIN.IGG.DELETE.NOSCRTCH
STGADMIN.IGG.DELGDG.FORCE
STGADMIN.IGG.DELNVR.NOBCSCHK
STGADMIN.IGG.DIRCAT

2. Password checking is bypassed if the function is performed on a system-managed data set.

In addition to the individual profiles, we recommend that the STGADMIN.* profile be defined with UACC NONE. Some STGADMIN profiles allow you to perform a specific function or use a specific keyword, but some functions or keywords can be used unless there is a STGADMIN profile preventing the usage of that keyword. By defining an STGADMIN.* profile with UACC NONE, these sensitive keywords can be protected and system exposures eliminated.

You need to define the following RACF profiles: [3]

STGADMIN.IDC.BINDDATA
Controls the ability to use the access method services BINDDATA command.

STGADMIN.IDC.DCOLLECT
Controls the ability to use the access method services DCOLLECT command.

STGADMIN.IDC.DIAGNOSE.CATALOG
Controls the ability to run the access method services DIAGNOSE command against catalogs.

STGADMIN.IDC.DIAGNOSE.VVDS
Controls the ability to run the access method services DIAGNOSE command against a VVDS when a comparison against the BCS is performed. In this case, the BCS is protected.

STGADMIN.IDC.EXAMINE.DATASET
Controls the ability to run the access method services EXAMINE command against integrated catalog facility catalog data sets.

STGADMIN.IDC.LISTDATA
Controls the ability to use the access method services LISTDATA command.

STGADMIN.IDC.LISTDATA.ACCESSCODE
Controls the ability to use the access method services LISTDATA ACCESSCODE command.

Note: Accesscode is a specialized LISTDATA command that requires an extra level of protection. Both levels (LISTDATA and LISTDATA.ACCESSCODE) are required.

STGADMIN.IDC.SETCACHE
Controls the ability to use the access method services SETCACHE command. This RACF profile does not include the following four profiles. A user must have SETCACHE access in order to have specific setcache command authorization.

STGADMIN.IDC.SETCACHE.DISCARDPINNED
Controls the ability to use the access method services SETCACHE DISCARDPINNED command.

STGADMIN.IDC.SETCACHE.PENDINGOFF
Controls the ability to use the access method services SETCACHE PENDINGOFF command.

STGADMIN.IDC.SETCACHE.REINITIALIZE
Controls the ability to use the access method services SETCACHE REINITIALIZE command.

---

3. See *DFSMS/MVS DFSMSdss Storage Administration Guide* for more information on DFSMSdss functions.

STGADMIN.IDC.SETCACHE.SUBSYSTEM
Controls the ability to use the access method services SETCACHE SUBSYSTEM command.

STGADMIN.IGD.ACTIVATE.CONFIGURATION
Controls the ability to activate an SMS configuration.

STGADMIN.IGG.ALTBCS
Controls the ability to alter BCS catalog attributes.

STGADMIN.IGG.ALTER.SMS
Controls the ability to alter the storage class and management class of a data set. If this profile is not created, the user must have RACF authority to the storage class and the management class in order to alter it.

STGADMIN.IGG.ALTER.UNCONVRT
Controls the ability to alter a system-managed VSAM data set to an unmanaged VSAM data set.

STGADMIN.IGG.DEFDEL.UALIAS
Controls the ability to define or delete an alias related to a user catalog without any other security authority. Note that you can still define or delete an alias if you have alter authority to the catalog, even if you do not have read authority to this facility class.

STGADMIN.IGG.DEFNVSAM.NOBCS
Controls the ability to define a non-VSAM data set with no BCS entry. Only a VVDS record (an NVR) for the system-managed non-VSAM data set is created.

STGADMIN.IGG.DEFNVSAM.NONVR
Controls the ability to define a non-VSAM data set with no VVDS entry (an NVR). Only a BCS entry for the system-managed non-VSAM data set is created.

STGADMIN.IGG.DELETE.NOSCRTCH
Controls the ability to delete the BCS entry for a system-managed data set without deleting the data set itself (for example, using DELETE NOSCRATCH). This controls functions which uncatalog data sets.

STGADMIN.IGG.DELGDG.FORCE
Controls the ability to use DELETE FORCE on a generation data group which contains a system-managed generation data set.

STGADMIN.IGG.DELNVR.NOBCSCHK
Controls the ability to delete the VVDS entry (the NVR) for an system-managed non-VSAM data set without checking the BCS entry and catalog name for the data set. If there is a BCS entry or if the catalog name contained in the NVR does not match the catalog provided in the request, the function is denied unless the user has authority to this profile.

STGADMIN.IGG.DIRCAT
Controls the ability to direct a catalog request to a specific catalog, bypassing the normal catalog search. A directed catalog request is one in which the catalog name is explicitly passed to the catalog in the CATALOG parameter of access method services commands.

In an SMS environment, all the catalog requests against system-managed data sets should be satisfied by the normal catalog search order. Directing the catalog request to a specific catalog requires authority to this profile with the exception of LISTCAT and EXPORT DISCONNECT requests.

STGADMIN.IGG.DLVVRNVR.NOCAT
: Controls the ability to delete a VVR or NVR without an associated catalog. Users having RACF READ authority to the facility class need no other RACF authority to the master catalog to perform the DELETE VVR or DELETE NVR functions. **CAUTION:**

  Access to this facility class should be restricted to users who understand the risk involved in deleting a VVR or NVR entry from a VVDS.

  When a catalog is deleted for recovery purposes, or under certain failure conditions, an uncataloged VSAM data set or SMS nonVSAM data set can be left on the volume. The user can issue the DELETE VVR or DELETE NVR command to clean up the volume. In order to do this, the user needs RACF ALTER authority to the master catalog, and the user catalog must exist so that the catalog can be searched to verify that a BCS entry does not exist for the VVR or NVR. This is the usual situation when RACF ALTER authority to the catalog is needed. If the user catalog does not exist, the user must define an empty user catalog so that it can be searched.

  The STGADMIN.IGG.DLVVRNVR.NOCAT facility class allows the use of DELETE VVR or DELETE NVR without an associated user catalog. It does not require RACF authority to the master catalog for these commands.

STGADMIN.IGG.LIBRARY
: Controls the ability to DEFINE, DELETE, or ALTER library and volume entries in a tape library.

STGADMIN.IGWSHCDS.REPAIR
: Controls the ability to use the AMS SHCDS command functions, which you can use to list outstanding SMSVSAM recovery and control that recovery.

Authority to Activate a Storage Management Subsystem Configuration

The FACILITY resource class profile STGADMIN.IGD.ACTIVATE.CONFIGURATION protects the ability to activate an SMS configuration. If you issue the ACTIVATE command from the Control Data Set Application Selection panel of ISMF, and either the Facility Resource Class is inactive or the named profile does not exist, the operator is queried to decide whether the ACTIVATE action should be allowed. You remain on the panel without the ability to provide additional input until action is taken from the operator console. If you have authority to activate an SMS configuration from ISMF, activation proceeds without confirmation from the operator console, and you continue with normal operations.

Storage Administrator Facility Class Profiles for DFSMSdss

Your installation can define one or more of the following special DFSMSdss storage administrator FACILITY class profiles: [4]

STGADMIN.ADR.CONVERTV
: Controls the ability to use the DFSMSdss CONVERTV command.

4. See *DFSMS/MVS DFSMSdss Storage Administration Guide* for more information on DFSMSdss functions.

STGADMIN.ADR.COPY.BYPASSACS
Controls the ability to have the DFSMSdss COPY command bypass the ACS routines and use the original or user-specified storage class and management class.

STGADMIN.ADR.COPY.INCAT
Controls the ability to use the INCAT parameter on a DFSMSdss COPY command.

STGADMIN.ADR.COPY.PROCESS.SYS
Controls the ability to process system data sets with the DFSMSdss COPY command.

STGADMIN.ADR.DUMP.INCAT
Controls the ability to use the INCAT parameter on a DFSMSdss DUMP command.

STGADMIN.ADR.DUMP.PROCESS.SYS
Controls the ability to process system data sets with the DFSMSdss DUMP command.

STGADMIN.ADR.RELEASE.PROCESS.SYS
Controls the ability to process system data sets with the DFSMSdss RELEASE command.

STGADMIN.ADR.RESTORE.BYPASSACS
Controls the ability to have the DFSMSdss RESTORE command bypass the ACS routines and use the original or user-specified storage class and management class.

STGADMIN.ADR.STGADMIN.COMPRESS
READ access to this profile permits you to perform a COMPRESS without needing UPDATE access authority to the data sets being compressed.

STGADMIN.ADR.STGADMIN.COPY
READ access to this profile permits you to perform COPY data without having READ or UPDATE access authority to the data sets being copied. This includes the ability to read the catalog for the source data set and update the catalog for the target data set.

STGADMIN.ADR.STGADMIN.COPY.DELETE
READ access to this profile permits you to specify the DELETE parameter with the COPY command and the ADMINistrator parameter. With this, you can delete a data set without having to have ALTER access authority to the data set or ALTER access authority to the catalog for that data set. RACF can still be called in order for DFSMSdss to determine if erase-on-scratch processing for a given data set needs to be performed.

STGADMIN.ADR.STGADMIN.COPY.RENAME
READ access to this profile permits you to specify the RENAME or RENAMEU parameter with the COPY command and the ADMINistrator parameter. With this, you can copy the data set with a new name without having to have sufficient access authority to the new name for the target data set or to the catalog for the target data set.

STGADMIN.ADR.STGADMIN.DEFRAG
READ access to this profile permits you to perform a DEFRAG without having READ authority to the data sets that are moved. RACF can still be called in order for DFSMSdss to determine if erase-on-scratch processing for a given data set needs to be performed.

STGADMIN.ADR.STGADMIN.DUMP
READ access to this profile permits you to DUMP data without having READ access authority to the data sets being dumped. This includes the ability to read the catalog for the source data set.

STGADMIN.ADR.STGADMIN.DUMP.DELETE
READ access to this profile permits you to specify the DELETE parameter with the DUMP command and the ADMINistrator parameter. With this, you can delete a data set without having ALTER access authority to the data set or ALTER access authority to the catalog for that data set. RACF can still be called in order for DFSMSdss to determine if erase-on-scratch processing for a given data set needs to be performed.

STGADMIN.ADR.STGADMIN.PRINT
READ access to this profile permits you to print data without having READ access authority to the data being printed.

STGADMIN.ADR.STGADMIN.RELEASE
READ access to this profile permits you to RELEASE allocated but unused space without having UPDATE access authority to the data sets.

STGADMIN.ADR.STGADMIN.RESTORE
READ access to this profile permits you to RESTORE data without having READ, UPDATE, or ALTER access authority to the data being restored. This includes the ability to read the catalog for the source data set and update the catalog for the target data set.

STGADMIN.ADR.STGADMIN.RESTORE.RENAME
READ access to this profile permits you to specify the RENAME or RENAMEU parameter with the RESTORE command and the Administrator parameter. With this, you can restore the data set with a new name without having sufficient access authority to the new name for the target data set or to the catalog for the target data set.

FIELD Resource Class

The RACF FIELD resource class controls the ability of users to specify or update the following RACF profile fields:

- Resource owner, RESOWNER, for data set profiles
- Default SMS DATACLAS, STORCLAS, and MGMTCLAS values for user/group profiles
- Application identifier, DATAAPPL for user/group profiles

How Authorization and Protection of Classes Differ

The *authorization* of SMS classes differs from the *protection* of SMS classes. For example, if you withhold authorization for a particular storage class, end users cannot specify that storage class in their JCL stream. End users specifying the storage class receive a JCL error if the ACS routine doesn't override the error.

Protection involves more than using an SMS class, it involves changing one. All end users might have READ access to the data set containing a particular storage class, but only a select few have UPDATE authority. Also, only people with UPDATE authority to the SCDS can update the SMS classes.

Authorizing the Storage and Management Classes

You can authorize the use of storage classes and management classes. Data classes do not require authorization.

Two of RACF's resource classes are: STORCLAS and MGMTCLAS. You authorize storage classes and management classes by defining them as RACF profiles to the general resource classes. The two resource classes, storage class and management class, are distinct from SMS storage class and management class.

Protecting the Control Data Sets

You need to protect three types of SMS data sets:

- Control data sets
    SCDS
    ACDS
    COMMDS
- ACS Routine Source Data Sets
    - Partitioned data sets whose members are source ACS routines
    - Sequential data sets that contain one source ACS routine.
- ACS Routine test library
    Partitioned data sets whose members are ACS test cases.

You can issue SECURITY from the command line of the ISMF Data Set Application to protect these data sets. You can give universal read access to ACDS, COMMDS, and SCDS to anyone. Anyone who is authorized to update an SCDS or an ACS routine source data set is also authorized to change any items within these data sets.

Authorizing for TSO and ISPF

To define an SMS data set under TSO, you do not need APF authorization, but it is recommended because APF authorization reduces path length.

For the DEFINE and IMPORT commands, you can establish this authorization by adding the DEFINE and IMPORT command processor names to the installation access list APFCTABL in CSECT IKJEFTE2, or the parameter AUTHCMD NAMES in SYS1.PARMLIB member IKJTSO00. For a call of access method services, you can establish this authorization by adding IDCAMS to the installation access list APFPTABL in CSECT IKJEFTE8 or the parameter AUTHPGM NAMES in SYS1.PARMLIB member IKJTSO00. For more information about specifying authorized commands and programs under TSO, see *OS/390 TSO/E Customization*.

To define or import an SMS data set while under option 6 of ISPF/PDF, you must include the DEFINE and IMPORT processor names in the command table in ISPTCM. See *OS/390 ISPF Planning and Customizing* for more information.

Storage Administrator Authorization

Storage administrators should receive the following authority from the security administrator:

- CLAUTH to STORCLAS and MGMTCLAS resource classes
- Authority to activate an SMS configuration
- Authority to perform exceptional catalog and DFSMSdss operations on system-managed data sets
- Authority to all ISMF applications, dialogs, line operators, and commands
- Update authority to the RESOWNER field in the DATASET profiles
- Update authority to the DATAAPPL, MGMTCLAS, STORCLAS, and DATACLAS, fields in the USER/GROUP profiles
- Authority to alter SMS constructs.

It might be preferable to have a group profile with authorities mentioned above for all storage administrators.

Chapter 14. Administering VSAM Record-Level Sharing

VSAM record-level sharing (RLS) is a data set access mode that allows multiple address spaces, CICS application owning regions (AORs) on multiple MVS systems, and jobs to access data at the same time. With VSAM RLS, multiple CICS systems can directly access a shared VSAM data set, eliminating the need for function shipping between application owning regions (AORs) and file owning regions (FORs). CICS provides the logging, commit, and rollback functions for VSAM recoverable files; VSAM provides record-level serialization and cross-system caching. CICS, not VSAM, provides the recoverable files function.

This chapter assumes you are familiar with the concepts of VSAM RLS and how it uses the coupling facility (CF) cache and lock structures. It describes the following tasks to enable and maintain VSAM RLS:

- “Preparing for VSAM Record-Level Sharing”
- “Defining CF Cache Structures in the SMS Base Configuration” on page 241
- “Defining Storage Classes for VSAM RLS” on page 244
- “Activating VSAM RLS” on page 246
- “Monitoring the Coupling Facility for VSAM RLS” on page 247
- “Recovering VSAM RLS Processing” on page 253
- “Falling Back from VSAM RLS Processing” on page 254

For more information on VSAM RLS, see the following publications:

- *DFSMS/MVS Planning for Installation*
- *DFSMS/MVS Using Data Sets*
- *OS/390 Parallel Sysplex Hardware and Software Migration*
- *OS/390 Parallel Sysplex Application Migration*
- *CICS Transaction Server for OS/390 Migration Guide*

Preparing for VSAM Record-Level Sharing

Planning for and installing VSAM RLS requires coordination with system hardware and software groups. This section describes the tasks that the storage administrator must do with the MVS systems programmer and the CICS database administrator to prepare to use VSAM RLS:

- “Determining Hardware Requirements” on page 230
- “Understanding the Product Environment for VSAM RLS” on page 230
- “Determining Which Applications Can Use VSAM RLS” on page 231
- “Ensuring Same Systems Connectivity” on page 232
- “Planning for Availability” on page 233
- “Defining Sharing Control Data Sets” on page 233
- “Defining CF Cache Structures” on page 235
- “Defining the CF Lock Structure” on page 237
- “Modifying the SYS1.PARMLIB IGDSMSxx Member” on page 241
- “Establishing Authorization for VSAM RLS” on page 241

See *DFSMS/MVS Planning for Installation* for a more detailed discussion of these tasks.

Determining Hardware Requirements

We recommend that you use multiple CFs with global connectivity. This ensures maximum availability. It also simplifies systems management and allows for non-disruptive lock transfer in the event of a CF outage.

You must have at least one CF connected to all systems capable of VSAM RLS within the Parallel Sysplex. For multiple CFs, select one facility with global connectivity to contain the master lock structure. For maximum availability, you should use a second CF with global connectivity. This way, in the event a CF becomes unavailable, the SMSVSAM address space can transfer its in-storage duplexed copy of the locks to the other available CF without causing any application disruption.

You can attach CFs which do not contain the master lock structure to a subset of the systems. A CF cache structure which is referenced by a storage class cache set must have at least the same connectivity as the storage groups to which the storage class maps.

Understanding the Product Environment for VSAM RLS

VSAM RLS processing involves support from multiple products: CICS, CICSVR and DFSMS/MVS. When either falling back from or moving forward to the CICS Transaction Server, CICSVR 2.3, DFSMS/MVS 1.3, and DFSMS/MVS 1.4, a number of rules apply. This section summarizes those rules and also describes some general considerations for the SMS environment.

Some of the general rules to consider when falling back from or moving forward to CICS, CICSVR or DFSMS/MVS release levels are:

- You *must* take image copies when falling back from CICSVR 2.3 to 2.2.
- You must take care of all backouts and indoubt resolutions on the current level of CICS/CICSVR before falling back or moving forward to other levels.
- You must delete any RLS information in the catalog when falling back from DFSMS/MVS 1.3 or 1.4 to DFSMS/MVS 1.2.
- When you delete RLS information in the catalog, you must revert to CICS FCT definitions, even if the CICS level remains unchanged.
- You must use the DEFINE CLUSTER or ALTER CLUSTER command to create RLS information in the catalog when moving forward from DFSMS/MVS 1.2 to DFSMS/MVS 1.3 or 1.4, if you intend to use VSAM RLS processing.

There are also some considerations specific to SMS SCDSs:

- You must apply toleration PTFs to all pre-DFSMS/MVS 1.3 systems in your Parallel Sysplex so that these systems can tolerate the changes in the SCDS format for DFSMS/MVS 1.3 and 1.4. You can only share SCDSs among DFSMS/MVS 1.3 or above systems running in compatibility mode and other down-level DFSMS/MVS systems.
- To obtain any RLS functions, such as cache sets, you must use ISMF on a DFSMS/MVS 1.3 or above system. The SCDS altered by ISMF can be activated from any system.

For more information on CICS and CICSVR, see the *CICS Recovery and Restart Guide*. See *DFSMS/MVS Planning for Installation* for more information on coexistence among products and product levels, as well as information on maintaining toleration among DFSMS/MVS 1.3 or 1.4 and lower-level systems. For the detailed procedure for falling back from VSAM RLS processing, see "Falling Back from VSAM RLS Processing" on page 254.

Determining Which Applications Can Use VSAM RLS

Applications using VSAM RLS benefit from increased data availability inherent in a shared environment that has read integrity and record-level locking (as opposed to control interval (CI) locking).

Your applications should fall into one of the following categories:

VSAM RLS-tolerant application
The application runs correctly in a multi-update environment when RLS is specified in the JCL or when MACRF=RLS is specified on the ACB. The RLS JCL parameter allows batch read programs to use RLS without requiring a recompile of the program. For batch update programs running against non-recoverable VSAM RLS spheres, it might be possible to modify the allocation from DISP=OLD to DISP=SHR.

VSAM RLS-exploiting application
The application recognizes when a VSAM data set can be shared at the record level and uses VSAM RLS functions to access the data. CICS applications are exploiting applications.

VSAM RLS-intolerant application
The application uses facilities not supported by VSAM RLS, accesses VSAM internal data structures, or is incompatible with the functions of VSAM RLS. For example, the application might perform CI access against the data set.

Because the CICS recoverable files function provides transactional recovery for applications, VSAM RLS is expected to be used primarily by CICS applications. Transactional recovery isolates the changes made by each sharing application. CICS creates a backout log record for each change made to a recoverable file, and VSAM RLS obtains and holds a lock on each changed record. The lock remains held until the transaction ends. If a transaction fails, CICS backs out all changes made by the application to recoverable files, thus isolating the other sharing applications from the failure.

In addition to the data sharing across CICS applications, VSAM RLS enables read-with-integrity sharing by batch jobs. Batch jobs can share recoverable files while they are being modified by CICS applications. This is possible because VSAM provides the record locking and buffer coherency functions across CICS and batch. Since VSAM RLS does not provide the transactional recovery function, it does not allow a batch job to open a recoverable data set for output.

VSAM RLS permits read and write sharing of non-recoverable data sets across CICS and batch jobs. Transactional recovery does not apply to non-recoverable data sets. While VSAM RLS permits sharing, the jobs must be carefully designed to achieve correct results in a read/write data sharing environment, since they do not have the isolation provided by transactional recovery.

Ensuring Same Systems Connectivity

You must ensure same systems connectivity for CF cache structures, lock structure, and storage groups. This is to ensure that jobs running in the Parallel Sysplex have access to data in both the CFs and storage groups. The lock structure must have global connectivity to all systems in the Parallel Sysplex.

Figure 105 shows how connectivity among systems and storage groups matches connectivity among systems and coupling facilities.
Figure 106 illustrates the problems that can occur if you do not have global

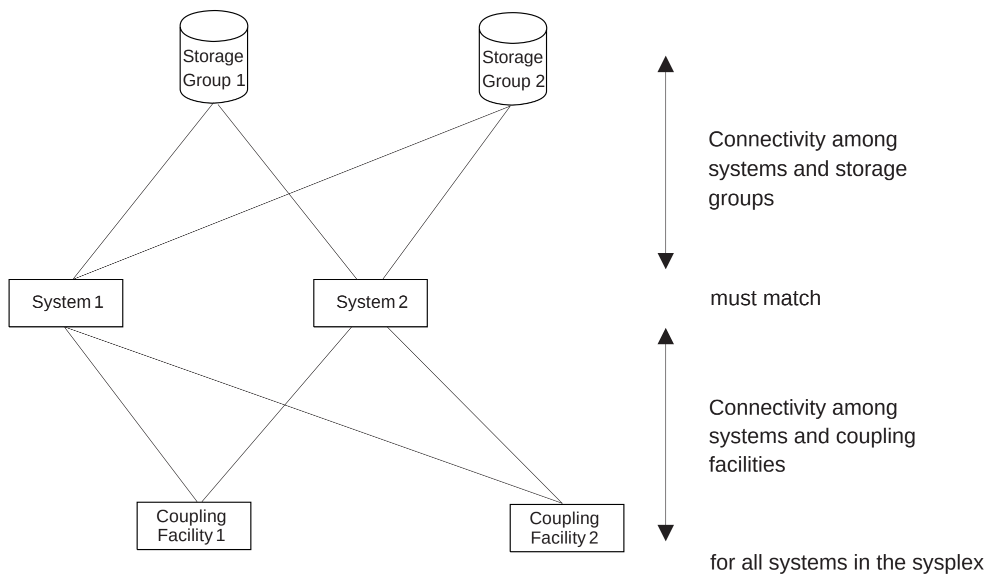

Figure 105. Example of Global Connectivity in a Parallel Sysplex

connectivity across the Parallel Sysplex:
In Figure 106, System 3 is connected to both Storage Group 2 and to Coupling

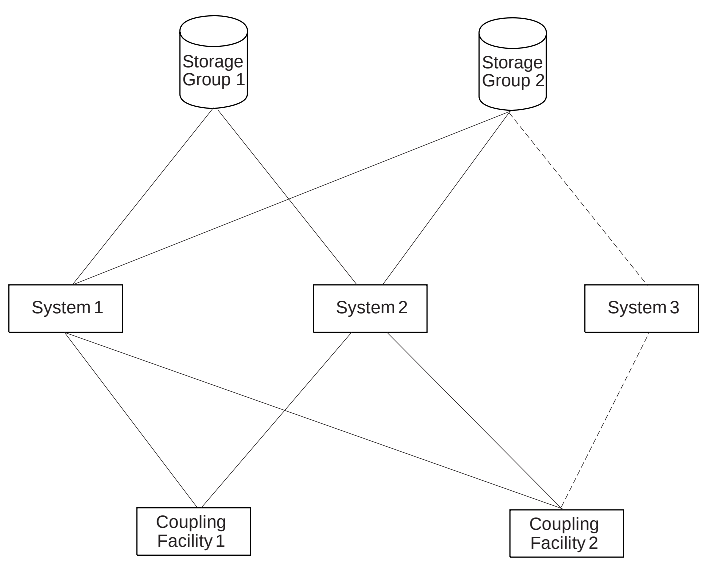

Figure 106. Example of Insufficient Connectivity in a Parallel Sysplex

Facility 2. This meets the minimum connectivity requirements but can still cause problems. Since Storage Group 2 data can be placed in Coupling Facility 1, jobs in

System 3 can fail if they attempt to access data sets in Storage Group 2 that have been assigned to Coupling Facility 1. You can avoid such a scenario by ensuring that, for all systems in the Parallel Sysplex, connectivity among systems and storage groups matches connectivity among systems and coupling facilities.

Planning for Availability

In order for VSAM RLS processing to take place, you must ensure the following:

- Run all systems performing RLS as a Parallel Sysplex
- Define and activate at least two sharing control data sets (SHCDS), and one spare SHCDS for recovery purposes
- Define CF cache and lock structures to MVS, using the CF resource manager (CFRM) policy. Define CF cache structures in the SMS base configuration.
- Associate CF cache set names with storage class definitions, and write ACS routines to associate storage class definitions that map to CF cache structures with data sets.
- Change the attributes for a data set to specify whether the data set is to be recoverable or non-recoverable. Specify LOG(NONE) if the data set is non-recoverable; specify LOG(UNDO) or LOG(ALL) if the data set is recoverable.

See *DFSMS/MVS Using Data Sets* for more information.

Defining Sharing Control Data Sets

Sharing control is a key element in maintaining data integrity in a shared environment. Because persistent record locks are maintained in the CF, several new classes of failure could occur, such as a Parallel Sysplex, system, or SMSVSAM address space restart, or a CF lock structure failure. The sharing control data set (SHCDS) is designed to contain the information required for DFSMS/MVS to continue processing with a minimum of unavailable data and no corruption of data when failures occur.

The SHCDS acts as a log for sharing support. It is a logically-partitioned linear data set that can be defined with secondary extents, though all extents for each data set must be on the same volume. An SHCDS contains the following information:

- The name of the CF lock structure in use
- The system status for each system or failed system instance
- The time that the system failed
- A list of subsystems and their status
- A list of open data sets using the CF
- A list of data sets with unbound locks
- A list of data sets in permit non-RLS state

Sharing control is critical to maintaining data integrity in the event of the loss of the lock structure. You should always have at least two active and one spare SHCDSs. Place these data sets for maximum availability. If necessary, you can have up to five active SHCDSs and five spare SHCDSs. Because the contents of these data sets are highly dynamic, we do not recommend backup and restore functions, since these could cause the loss of VSAM RLS data set recovery information.

Consider the following as you allocate and maintain your SHCDSs:

- Allocate SHCDSs so that the number of active and spare data sets ensures the data is always duplexed. At a minimum, define and activate two SHCDSs and at least one spare SHCDS for recovery purposes. You should ensure that there are enough spare SHCDSs, since these are used when I/O errors occur on the active SHCDSs.

- Place the SHCDSs on volumes with global connectivity. VSAM RLS processing is only available on those systems that currently have access to the active SHCDS. The share options for SHCDSs must be set to (3,3) so that each system in the Parallel Sysplex can properly share the data sets. See *DFSMS/MVS Access Method Services for VSAM* for more information on share options.
- Place your SHCDSs in such a way as to maximize availability in the event of the loss of a volume. Use storage classes defined with the guaranteed space attribute. Avoid placing SHCDSs on volumes for which there might be extensive volume reserve activity.
- Ensure that the space allocation for active and spare SHCDSs is the same.
- Use the VARY SMS,SHCDS and DISPLAY SMS,SHCDS commands to maintain your SHCDSs.

You must use the following naming convention when defining your SHCDSs:

SYS1.DFPSHCDS.*qualifier***.V***volser*

where:

qualifier
: is a 1 to 8 character qualifier.

volser
: is the volume serial number. The V prefix allows you to specify numeric volume serial numbers.

Use the following formula to calculate the size of your SHCDS:

```
Space = (16 + (number_of_systems x (16 + number_of_OPENs/10))) kilobytes
```

where:

Space
: is the space required for the SHCDS.

number_of_systems
: is the number of systems.

number_of_OPENs
: is the number of concurrent OPENs to the CF that you expect.

For example, if you have 3 systems and expect to have 2000 concurrent OPEN requests, your SHCDS requires 664 KB of space.

You can create an SHCDS using JCL, or using access method services or TSO subcommands. When you use the access method services DEFINE command to create an SHCDS, specify SHAREOPTIONS(3,3) to ensure that the SHCDS can be written to and read from any system.

The following shows an example of how to create an SHCDS using JCL:

Note: When you use JCL, the default for share options is (1,3). If you use JCL to define the SHCDSs, use a data class defined with a cross-region share options value of 3 and a cross-system share options value of 3.

```
//PRISHCDS DD DSN=SYS1.DFPSHCDS.PRIMARY.VSMS001,SPACE=(1,(10,10)),
//            RECORG=LS,STORCLAS=GSPACE,VOL=SER=SMS001,AVGREC=M,
//            DISP=(NEW,CATLG)
//SECSHCDS DD DSN=SYS1.DFPSHCDS.SECONDRY.VSYS002,SPACE=(1,(10,10)),
//            RECORG=LS,VOL=SER=SYS002,AVGREC=M,
//            DISP=(NEW,CATLG)
//SPRSHCDS DD DSN=SYS1.DFPSHCDS.SPARE.VSMS003,SPACE=(1,(10,10)),
//            RECORG=LS,STORCLAS=GSPACE,VOL=SER=SMS003,AVGREC=M,
//            DISP=(NEW,CATLG)
```

Figure 107. JCL Example for Creating Sharing Control Data Sets

Once the SHCDS data sets have been created, you make them available for use by using the VARY SMS(...),NEW command. This command only needs to be entered on one system in the Parallel Sysplex. The SMSVSAM address space on the system where the command is entered communicates the name of the data set to the other SMSVSAM address spaces in the Parallel Sysplex. Each SMSVSAM address space recatalogs the data set, if it has not already been cataloged, so that you do not need to manually catalog the data set in order for it to be used. Those data sets added for use are saved, and can be accessed when an SMSVSAM address space initializes them or is restarted, or though subsequent IPLs.

Defining CF Cache Structures

CF cache structures must be defined to MVS/SP and also in the SMS base configuration. CF cache structures provide a level of storage hierarchy between local memory and DASD cache. They are also used as a system buffer pool for VSAM RLS data when that data is modified on other systems. Each CF cache structure is contained in a single CF. You might have multiple CFs and multiple CF cache structures.

Several factors determine the number and size of your CF cache structures:

- Number of available CFs
- Amount of space available in each CF
- Amount of data to be accessed through each CF
- Continuous availability requirements for CF reconfiguration
- Performance requirements for various applications

You use CFRM policy definitions to specify an initial and maximum size for each CF cache structure. DFSMS uses the initial structure size you specify in the policy each time it connects to a CF cache structure.

You can assign one or more CF cache structures to each cache set associated with a storage class. Having multiple cache sets allows you to provide different performance attributes for data sets with differing performance requirements. When more than one CF cache structure is assigned to a cache set, data sets are assigned to each CF cache structure in an effort to balance the load.

Determining CF Cache Structure Size

A CF cache structure must be at least large enough to hold all of the MVS information required to describe a structure of maximum size. To help you achieve the best possible performance with VSAM RLS buffering, the sum total of all the CF cache structure sizes you define (the CF cache) should ideally be the sum total of the local VSAM local shared resources (LSR) buffer pool sizes. The size of the local VSAM LSR buffer pool is the sum of LSR pool size and, if used, the corresponding hiperspace pool size. You can run VSAM RLS with less CF cache storage than this, but the CF cache *must* be large enough for the CF cache directories to contain an entry for each of the VSAM RLS local buffers across all instances of the SMSVSAM server. If the CF cache cannot contain the directory entries describing the local buffers, then the VSAM RLS local buffers are falsely invalidated and must be refreshed. To minimize this, the minimum CF cache structure size should never be less than 1/10 the size of the local buffer pool.

For example, the following CICS FOR configuration shows the sum total of the local VSAM RLS buffer pool size prior to migrating to VSAM RLS.

File Owning Region	LSR pool size	Hiperspace pool size	Sum Total
FOR_1	20MB	30MB	50MB
FOR_2	40MB	no pool	40MB
FOR_3	30MB	50MB	80MB
			170MB

When migrating this configuration to VSAM RLS, the CF cache you define should ideally be at least 170MB. In this way, cross-invalidated local RLS buffers can be refreshed from the CF cache structures.

Performance should improve when the CF cache is larger than the sum of the local VSAM LRS buffer pool sizes. When the CF cache is smaller, performance depends upon the dynamics of the data references among the systems involved. In some cases, you might want to consider increasing the size of very small CF caches (2MB to 10MB).

In those situations where you can determine that data previously treated as a non-shared resource (NSR) is no longer to be treated as such, you should also include NSR buffer sizes in the total local buffer pool size.

See *CICS Transaction Server for OS/390 Migration Guide* for more information on calculating a cache structure size.

Using the RLS_MAX_POOL_SIZE Parameter to Limit Local Buffer Pool Size

You can use the RLS_MAX_POOL_SIZE parameter of the IGDSMSxx parmlib member to limit the maximum size of the local buffer pool(s). The value you specify can be either larger or smaller than the default maximum pool size of 100MB, but it must be supported with the available real and expanded storage. Although VSAM RLS can in some cases ascertain how much buffer space is supported before paging begins to occur, specifying an RLS_MAX_POOL_SIZE value ensures that the local buffer pool does not grow beyond the value you specify. Note that in some instances, the local buffer pool might *temporarily* grow larger than the RLS_MAX_POOL_SIZE value. Note also that setting a RLS_MAX_POOL_SIZE value that is too low might result in unnecessarily degrading the local hit rate.

We recommend that you initially set the RLS_MAX_POOL_SIZE value to 50% more than the sum of the local buffers on a single system. The following table illustrates this, assuming that each CICS FOR is on a separate system:

File Owning Region	LSR pool size	Hiperspace pool size	Sum Total	RLS_MAX_POOL_SIZE
FOR_1	20MB	30MB	50MB	75MB
FOR_2	40MB	no pool	40MB	60MB
FOR_3	30MB	50MB	80MB	120MB

You can use the information from the SMF type 42 record with subtype 19 to evaluate the local buffer hit rates for each of the individual systems. Use this information together with the local system paging rates to help you make additional tuning adjustments to the RLS_MAX_POOL_SIZE parameter.

Defining the CF Lock Structure

To use VSAM RLS, you must define a single, master CF lock structure. For maximum availability, we recommend defining a non-volatile lock structure. The CF lock structure is used to enforce the protocol restrictions for VSAM RLS data sets, and to maintain the record-level locks and other DFSMSdfp serializations. You should ensure that the CF lock structure has universal connectivity, so that it is accessible from all systems in the Parallel Sysplex that support VSAM RLS.

The CF lock structure is named IGWLOCK00. Use the XES coupling definition process to define it. To estimate its size requirements in megabytes, use the following formula (note that a megabyte is 1048576 bytes in this case):

```
10M * number_of_systems * lock_entry_size
```

where:

number_of_systems
: is the number of systems in the Parallel Sysplex

lock_entry_size
: is the size of each lock entry. This value depends on the MAXSYSTEM value that is specified to the IXCL1DSU Couple Data Set format utility.

  Use the following table to determine the actual lock entry size for the different MAXSYSTEM setting values:

Table 16. Effect of MAXSYSTEM Value on Lock Table Entry Size

MAXSYSTEM Value	Lock Entry Size
7 or less	2 bytes
>= 8 and < 24	4 bytes
>= 24 and <= 32	8 bytes

Table 17 shows some sample lock allocation estimates:

Table 17. CF Lock Structure Sizing Examples

MAXSYSTEM Value	Number of systems	Total Lock Size
<= 7	2	40MB
	4	80MB
<= 23	8	320MB
<default> =8	2	80MB
	4	160MB
	8	320MB

Table 17. CF Lock Structure Sizing Examples (continued)

MAXSYSTEM Value	Number of systems	Total Lock Size
32	2	160MB
	4	320MB
	8	320MB

These lock size estimates include the memory requirements for both the lock table and the record-lock memory. Use these estimates as rough initial values to help you attain a locking structure with a desired false contention target of approximately one-half of 1% or less. Contact your marketing representative for help in arriving at an initial estimate that more closely matches your specific configuration.

Considerations for Retained Locks and Record Table Full Conditions

The CF lock structure includes two parts:

- a lock table, used to determine whether there is R/W interest among systems on a particular resource
- record table space to keep track of information for retained locks and spheres which have been processed by VSAM RLS

If a commit protocol application such as CICS fails, the locks protecting updates against recoverable spheres are remembered in the record table space until CICS performs the required backouts. Also, update locks associated with indoubt transactions are remembered until the indoubts are resolved.

When used record table space reaches 80% or greater, informational message, IGW326I, is issued. A shortage of record table space can occur for the following reasons:

- The size of the lock structure is too small for normal system operation.
- A CICS system has failed and cannot be successfully restarted to run its backouts. As a consequence, record table space cannot be freed up.
- Outstanding indoubt transactions exist. CICS provides commands to display indoubt transactions and resolve them.
- Transactions are in backout failed state. This means that the backouts could not complete and free record table space.

Since it might be difficult to remedy these situations quickly, you should respond to a record table shortage by modifying the CFRM policy to increase the size of the lock structure, in order to provide additional record table space and increase the space available for retained locks. Then, activate that policy and rebuild the lock structure via operator command.

You can increase record table space by carefully selecting the total lock memory allocation quantity. Record table space is guaranteed to never be less than one-half of the total lock space allocated to RLS, but it can be more if the total size is not a power of two. A lock space allocation of 32MB causes 16MB to be allocated to the lock table and 16MB to be allocated to the record lock table. An allocation of 63MB, on the other hand, causes 16MB to be allocated to the lock table, and all of the remaining memory (i.e. 47MB) is then allocated to the record table space. Thus, selecting a total lock size that is not a power of two value is a means of causing the record space to grow without necessarily increasing the table size.

If the record table becomes full, VSAM requests which require that a lock be recorded receive RPL feedback, and CICS backs out the transaction. A record table full condition means that transactions cannot complete successfully, and the size of the lock structure *must* be increased.

Avoiding False Contention

VSAM RLS assigns locked resources to an entry value in the lock table, and uses this entry value to quickly check whether a resource is already locked. If the lock structure (and thus the lock table) is too small, many locks can be represented by a single value, making ″false″ lock contention possible. False lock contention occurs when two different locks on different resources attempt to use the same lock entry. The second lock requester is suspended until VSAM RLS determines that there is no real lock contention on the resource. False contention can be a problem for workloads with heavy R/W interests among systems.

To avoid false contention, you need to consider the size of the lock table. The lock table size is determined by the total size of the lock structure. When you define the size of the total lock structure, you should specify a value that is a power of two in order to maximize the lock table size: the lock table comprises 50% of the total space, and the record lock space the remaining 50%. Any memory in excess of a power of two value is allocated to the record lock space exclusively until the next power of two value is reached. At that time, the lock table space is doubled, and the two allocations are once again of equivalent sizes.

VSAM RLS uses the MAXSYSTEM value from the Couple Data Sets format utility to determine the size of each lock entry. Since smaller lock entry sizes imply a larger number of locks for the same memory allocation, it is very important to select an appropriate MAXSYSTEM value. The MAXSYSTEM value represents the maximum number of systems that can be connected into the Parallel Sysplex. There is a lock table memory penalty when the MAXSYSTEM setting exceeds 7, and another penalty when it exceeds 23. Consequently, from a false contention point of view you want to select a MAXSYSTEM value that does not exceed 7 or 23.

To increase a MAXSYSTEM value that has already been specified, you must format larger CDSs, switch them into use dynamically, and manually rebuild the CF lock structure using the SETXCF START command.

To decrease a MAXSYSTEM value that has already been specified, you must shut down all of the VSAM RLS address spaces, and manually delete both the persistent connections and the lock structure. Then you must restart the VSAM RLS address spaces. If you decrease the MAXSYSTEM value without first shutting down, the decrease has no effect; the Parallel Sysplex continues to run using the old MAXSYSTEM value.

Monitoring for False Contention: You can determine the amount of false contention by using either the resource measurement facility (RMF) or the DISPLAY SMS,CFLS command.

How Much Contention is Acceptable: For the best performance, you want to achieve the least possible amount of global lock contention, both real and false. The amount of real lock contention is application-dependent; it depends on record access patterns. False lock contention is almost entirely determined by the size of the lock table, with a larger lock table having less false lock contention than a smaller one. A good goal is to have total (real and false) global lock contention of less than one percent. The false contention component of the total global lock contention should be less than one-half of one percent, and ideally, should be substantially less than this.

Reducing False Contention: If false contention becomes a problem, try the following:

- Reduce the amount of real lock contention in your applications, if possible
- Specify a larger size for the lock structure and manually rebuild it
- Ensure the MAXSYSTEM parameter of the Couple Data Set utility is not too large for the number of members in your Parallel Sysplex

  A MAXSYSTEM value of 7 or less allows you twice as many lock entries as a MAXSYSTEM value of 8, as shown in Table 16 on page 237.

Adjusting the Lock Structure Size

Once you select an initial lock structure size and the Parallel Sysplex has been running with that size for some time, you should monitor the percentage of false contentions. You can then use this percentage to help you select an even more appropriate lock structure size.

Use the following formula to determine the estimated lock structure size based on a false contention percentage:

```
Minimum Lock Structure Size = F * M / T
```

where:

F is the measured false contention percentage, expressed as a percentage (for example, a rate of 0.02 = 2%, so F = 2)

M is the current lock structure allocation size

T is the target false contention target percentage, expressed as a percentage (for example, a rate of 0.005 = 0.5%, so T = 0.5)

A Lock Structure Sizing Example

Suppose a Parallel Sysplex has a MAXSYSTEM setting of 3, with two systems currently connected to the CF. You could use the initial sizing formula to estimate its initial size, as shown below:

```
Initial Lock Structure Size = 10M * 2 * 2
```

where:

number_of_systems = 2

lock_entry_size = 2 bytes

This yields an initial lock structure size of 40MB. However, to maximize the lock table space itself, you should size it with a number that is a power of two. In this case, the initial total lock size could be set at either 32 or 64MB.

For the purpose of this example, we select 32MB as the initial lock structure size. We then run for a while before determining the rate of false contentions. Assuming that this false contention rate is 1.5%, and that we have a target false contention rate of 0.5%, we can then use the following formula to modify the lock structure size:

```
Minimum Lock Structure Size = 1.5 * 32MB / 0.5
```

where:

1.5 is the measured false contention rate of 1.5%

32MB is the specified lock structure size

0.5 is the target false contention rate of 0.5%

The adjusted size should now be 96MB. However, since this needs to be expressed as a power of two, unless we actually select a value of at least 128MB, it is unlikely that we will find that the false contention rate meets or exceeds our target of 0.5%.

In this example, the Parallel Sysplex would run better with a larger lock structure size than initially allocated. On the other hand, had the monitored false contention rate been less than an initial target value of 0.5%, it would not mean that we were wasting CF storage by having it allocated to the lock structure. In fact, although a false contention rate of 0.1% is not feasible in many cases, it is still ideal, assuming that it can be achieved with a reasonable amount of CF memory.

Modifying the SYS1.PARMLIB IGDSMSxx Member

The IGDSMSxx parmlib member includes the following parameters to support the CF and VSAM RLS processing:

- CF_TIME which is used to align creation of all the CF-related SMF type 42 records with subtypes 15, 16, 17, 18 and 19
- DEADLOCK_DETECTION which specifies the interval for detecting deadlocks between systems
- RLSINIT which specifies whether the SMSVSAM address space is started as part of system initialization or the V SMS,SMSVSAM,ACTIVE command
- RLS_MAX_POOL_SIZE which specifies the maximum size of the SMSVSAM local buffer pool
- SMF_TIME which is used to align the SMF type 42 records for DFSMS (with subtypes 1, 2, 15, 16, 17, 18 and 19) to the SMF_TIME interval

You can modify these parameters at any time during VSAM RLS processing. See “Changing IGDSMSxx Parameters to Support the Coupling Facility” on page 250 for information on changing these parameters.

Establishing Authorization for VSAM RLS

You want to establish the following authorization to restrict access to certain VSAM RLS capabilities:

- To use the access method services SHCDS command, you must be authorized to the facility class STGADMIN.IGWSHCDS.REPAIR. The SHCDS command is used to list SMSVSAM recovery associated with subsystems and spheres, and to control that recovery.
- Only those users who actually need the capability, such as CICS subsystems, should have access to register a subsystem name to SMSVSAM. Use the RACF subsystem name class to restrict this access. For more information, see *CICS Transaction Server for OS/390 Release Guide*.

Defining CF Cache Structures in the SMS Base Configuration

In order for DFSMSdfp to use the CF for VSAM RLS, after you define one or more CF cache structures to MVS, you must also add them in the SMS base configuration.

To add CF cache structures to the base configuration, you associate them with a cache set name. This cache set name is also specified in one or more storage class definitions. When a storage class associated with a data set contains a cache set name, the data set becomes eligible for VSAM record-level sharing and can be placed in one of the CF cache structures associated with the cache set. The system selects the best cache structure in which to place the data set.

See “Defining CF Cache Structures” on page 235 and “Defining the CF Lock Structure” on page 237 for more information on using CFRM policies to define CF cache and lock structures.

To define CF cache structures to DFSMS/MVS:

1. Select option 8 from the ISMF Primary Option Menu for Storage Administrators, to invoke the Control Data Set (CDS) Application Selection panel on ISMF.

   Figure 108 shows the CDS Application Selection panel:

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                          CDS APPLICATION SELECTION
Command ===>

To Perform Control Data Set Operations, Specify:
  CDS Name . . 'USER6.TESTCDS'
                            (1 to 44 Character Data Set Name or 'Active')

Select one of the following Options:

  7  1. Display       - Display the Base Configuration
     2. Define        - Define the Base Configuration
     3. Alter         - Alter the Base Configuration
     4. Validate      - Validate the SCDS
     5. Activate      - Activate the CDS
     6. Cache Display - Display CF Cache Structure Names for all CF Cache Sets
     7. Cache Update  - Define/Alter/Delete CF Cache Sets

If CACHE Display is chosen, Enter CF Cache Set Name . . *
                             (1 to 8 character CF cache set name or * for all)

Use ENTER to Perform Selection;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 108. CDS (Control Data Set) Application Selection for VSAM RLS

2. Specify the name of the SCDS that is to contain the base configuration for VSAM RLS in the CDS NAME field.

   When this panel is displayed, the CDS NAME field is primed with the last used SCDS name. This name might differ from the SCDS name of your base configuration for VSAM RLS.

3. Select option 7, CACHE UPDATE, and press ENTER. ISMF displays the CF Cache Set Update panel, shown in Figure 109 on page 243.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
DGTDBSA3                      CF CACHE SET UPDATE                  Page 1 of 1
Command ===>

SCDS Name  : USER6.TESTCDS
Define/Alter/Delete CF Cache Sets:       ( 000  Cache Sets Currently Defined )

 Cache Set                      CF Cache Structure Names
 ________  ________________ ________________ ________________ ________________
           ________________ ________________ ________________ ________________
 ________  ________________ ________________ ________________ ________________
           ________________ ________________ ________________ ________________
 ________  ________________ ________________ ________________ ________________
           ________________ ________________ ________________ ________________
 ________  ________________ ________________ ________________ ________________
           ________________ ________________ ________________ ________________
 ________  ________________ ________________ ________________ ________________
           ________________ ________________ ________________ ________________
 ________  ________________ ________________ ________________ ________________
           ________________ ________________ ________________ ________________
More CF Cache Sets to Add?  . . N  (Y/N)
Use ENTER to Perform Validation; Use UP/DOWN Command to View other Pages;
Use HELP Command for Help; Use END Command to Save and Exit.
```

Figure 109. Coupling Facility Cache Set Definition for VSAM RLS

4. Define your CF cache set(s).

   The CF Cache Set Update panel allows you to define up to 256 CF cache sets. Each CF cache set can have up to eight CF cache structure names assigned to it. Use the cache structure names you defined in the MVS CFRM policies.

Figure 110 on page 244 shows how CF cache structures can be defined in multiple cache sets, and also shows how data sets associated with a storage class definition containing a cache set name can be placed in multiple cache structures.

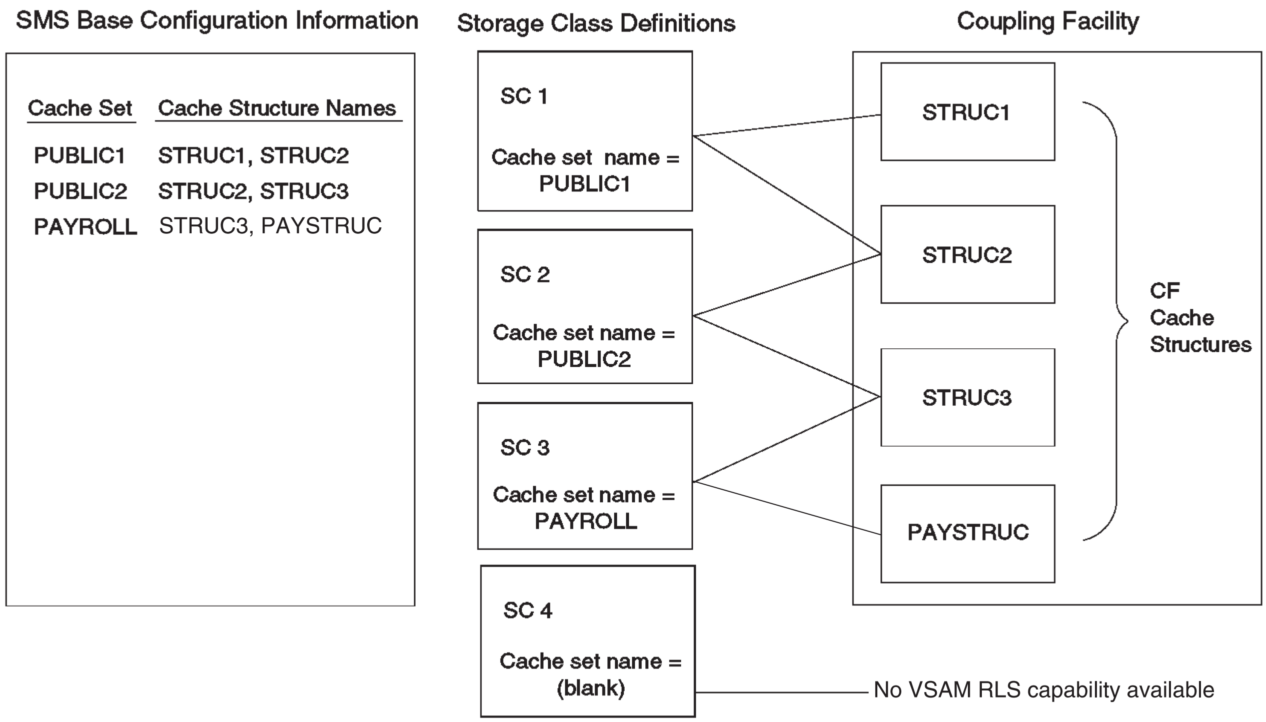

Figure 110. Example of CF Cache Structure Definition in Base Configuration

Defining Storage Classes for VSAM RLS

This section describes how to assign the cache set names defined in the base configuration to a storage class, so that data sets associated with that storage class can be eligible for VSAM RLS and use CF cache structures. It also describes how to indicate the relative importance of the data associated with the storage class.

Note: In a JES3 environment, be careful to define cache set names only in those SMS storage classes that are used by data sets opened for VSAM RLS processing. When you define a cache set name in a storage class, any job accessing a data set associated with that storage class is scheduled on a VSAM RLS-capable system. If all storage classes have cache set names defined for them, then all jobs accessing SMS-managed data sets are scheduled to VSAM-RLS-capable systems. This could cause a workload imbalance between those systems and down-level systems.

To assign the CF cache sets defined in the base configuration to storage classes, follow these steps:

1. Select option 5, STORAGE CLASS, from the ISMF Primary Option Menu for Storage Administrators. ISMF displays the Storage Class Application Selection panel, shown in Figure 111 on page 245:

```
  Panel  Utilities  Help
------------------------------------------------------------------------------
                     STORAGE CLASS APPLICATION SELECTION
Command ===>

To perform Storage Class Operations, Specify:
  CDS Name  . . . . . . . 'USER11.SCDS'
                               (1 to 44 character data set name or 'Active' )
  Storage Class Name  . . *         (For Storage Class List, fully or
                                    partially specified or * for all)
Select one of the following options  :
  3  1. List         - Generate a list of Storage Classes
     2. Display      - Display a Storage Class
     3. Define       - Define a Storage Class
     4. Alter        - Alter a Storage Class
     5. Cache Display - Display Storage Classes/Cache Sets
If List Option is chosen,
   Enter "/" to select option      Respecify View Criteria
                                   Respecify Sort Criteria

If Cache Display is Chosen, Specify Cache Structure Name . .

Use ENTER to Perform Selection;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 111. Defining a Storage Class for VSAM RLS

2. Specify the name of the SCDS you defined for VSAM RLS in the CDS NAME field.
3. Select option 3, DEFINE, and press ENTER. ISMF displays the Storage Class Define panel.
4. Press the DOWN key to view the second Storage Class Define panel, shown in Figure 112:

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                            STORAGE CLASS DEFINE                 Page 2 of 2
Command ===>

SCDS Name . . . . . : IBMUSER.TESTCDS
Storage Class Name  : STOR1

To  Storage Class, Specify:

  CF Cache Set Name . . . . . CSI         (up to 8 chars or blank)
  CF Direct Weight  . . . . . 6           (1 to 11 or blank)
  CF Sequential Weight  . . . 6           (1 to 11 or blank)







Use ENTER to Perform Verification; Use UP Command to View previous Page;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Figure 112. Defining Storage Class Attributes for VSAM RLS

5. Enter the name of the CF cache set you defined in the base configuration.

6. Specify a weight attribute for the data in the CF Direct Weight or the CF Sequential Weight fields to indicate the data’s relative importance. The default is a weight value of 6.

   Note: DFSMS/MVS only supports the default value. Regardless of what you specify, all data eligible for VSAM record-level sharing is assigned a weight value of 6.

Defining VSAM RLS Attributes in Data Classes

You can define data classes specifically for data sets eligible for VSAM RLS. When you have data classes with VSAM RLS parameters, you do not need to change current AMS DEFINE statements and job streams. Note that to avoid having too many data classes, you should specify the JCL LGSTREAM ID keyword for SMS VSAM data sets defined by JCL instead of the Logstream ID attribute in the data class. See “Defining Data Class Attributes” on page 116 for more information.

Activating VSAM RLS

The SMSVSAM address space always starts at IPL time, providing RLSINIT(YES) is specified in the IGDSMSxx parmlib member. VSAM RLS processing is available once certain minimum requirements are met. This section describes those requirements.

Enabling VSAM RLS Processing

VSAM RLS processing is available once the following requirements are met:

- All systems are running as a Parallel Sysplex.
- At least two SHCDSs, and one spare SHCDS have been activated.
- At least one existing CF cache structure is defined to MVS and to the SMS base configuration.

  If a subset of the cache structures is not available, some data sets might not be accessible.
- The CF lock structure IGWLOCK00 is available.
- The SMS address space is started.

If a null configuration is active, only existing data sets already assigned to a CF cache structure can be processed.

If VSAM RLS processing is not enabled, all attempts to open a data set where VSAM RLS is specified on the ACB or the JCL fail.

Under the following conditions, VSAM RLS access from a system is *not* available:

- No SHCDSs are available.
- The CF lock structure, IGWLOCK00, is not available.
- The SMSVSAM address space has failed and a response to message IGW418D is pending.
- The SMSVSAM address space has failed, and manual restart is in effect (the response to message IGW418D was `C`).

- None of the CF cache structures defined in the SMS configuration are available. If a subset of CF cache structures are not available, data sets bound to those structures might not be accessible from this system.

Enabling a Data Set for VSAM RLS Processing

For a data set to be opened for VSAM RLS processing, VSAM RLS processing must be available, and the LOG parameter must be specified on the DEFINE CLUSTER or the ALTER CLUSTER command for the data set. Additionally, if the data set is currently assigned to a CF cache structure, that cache structure must be available. If the data set is not currently assigned to a CF cache structure, a cache set must be specified on the storage class, and at least one of the associated CF cache structures must be available.

Monitoring the Coupling Facility for VSAM RLS

This section describes:
- Displaying information about the CF
- Altering the size of CF cache structures
- Altering the size of the CF lock structure
- Changing the IGDSMSxx parameters that support the CF
- Altering the Status of CF Cache Structures and Volumes
- Selecting Data Sets for CF Statistical Monitoring

Displaying CF Information

You can use MVS operator commands as well as ISMF to request CF information.
- Use the MVS DISPLAY XCF command to display information about CFs, connections to the CF, and CFRM policies.
- Use the DISPLAY SMS command or ISMF applications to request information about DFSMS CF structures and SHCDSs.

The following sections describe using the DISPLAY SMS command and ISMF applications in greater detail.

Using the DISPLAY SMS Command

The syntax for the DISPLAY SMS command is shown below.

DISPLAY SMS,SMSVSAM[,ALL] **,CFCACHE(***CF_cache_structure_name*|****)** **,CFLS** **,SHCDS** **,CFVOL(***volume_serial_number***)** **MONDS(***specification_mask***|*)**

The following list describes the various forms of the DISPLAY SMS command and what information is displayed:
- `DISPLAY SMS,SMSVSAM[,ALL]`

  displays the status of the SMSVSAM server. Specify ALL to see the status of all SMSVSAM servers.
- `DISPLAY SMS,CFCACHE(`*CF_cache_structure_name*`|*)`

displays information about CF cache structures. Specify CFCACHE(*) to request information for all CF cache structures. Specify a specific cache structure name to display information about only that cache structure.

- `DISPLAY SMS,CFLS`

  displays information about the CF lock structure. This information includes the lock rate, lock contention rate, false contention rate, and average number of requests waiting for locks.
- `DISPLAY SMS,SHCDS`

  displays information about SHCDSs. This information includes SHCDS names, sizes and the amount of free space for all the active SHCDSs and their status. It also includes the names of all the spare SHCDSs.
- `DISPLAY SMS,CFVOL(`*volser*`)`

  displays a list of CF cache structures containing data for the volume specified. Also displays the CF_VOLUME status.
- `DISPLAY SMS,MONDS(`*specification_mask*`|*)`

  Specify MONDS(*) to view all the data set specifications eligible for CF statistics monitoring. Use a specification mask to view only a subset of those specifications. You can specify a full or partial data set name, and you must specify at least one high level qualifier. A wild card in the data set name cannot be followed by additional qualifiers.

Using ISMF

You can use the ISMF Data Set, Volume and Storage Class applications to display CF information. You can also use ISMF to view the cache sets defined for a specific CDS and the cache structure names associated with each cache set.

Data Set Application: Use the Data Set Selection Entry panel to specify certain attributes and request a list of corresponding data sets and their characteristics. The data sets must be open for VSAM RLS access in order to appear on the list. If you specify CF attributes, ISMF displays the following CF information for each data set matching your criteria:

- CF status indicator

  Indicates whether the data set is in use by RLS processing, whether a forward recovery of the data set is in progress, and whether the sphere has been quiesced for RLS processing.
- CF monitor status

  Indicates whether CF cache structure statistical monitoring is on or off.
- CF cache structure name

  Specifies the name of the CF cache structure in which the data set is stored.
- CF cache set name

  Specifies the name of the CF cache set with which the data set is associated.

You can also use these values as attributes on the Data Set Filter, View, and Sort panels to customize the data set list created by ISMF.

Volume Application: Use the Volume Selection Entry panel to specify certain attributes and request a list of corresponding volumes and their characteristics. You can request that ISMF display CF volume status information, indicating one of the following conditions:

- The volume is enabled for VSAM RLS processing and can be associated with a CF cache structure

- VSAM RLS processing is finishing and no new data can be placed in CF cache structures
- No VSAM RLS data for the volume exists in any of the CF cache structures and no CF cache structures can be assigned to the volume

You can also use the CF volume status attribute on the volume filter, view, and sort panels to further customize the volume list created by ISMF.

Storage Class Application: Use the Storage Class Application Selection panel to request that ISMF display storage class names and CF cache set names associated with a specific CF cache structure name. You can also request to see the CF direct or sequential weights assigned to a specific storage class definition.

You can also use the CF cache set name, CF direct weight, and CF sequential weight attribute on the storage class filter, view, and sort panels to further customize the storage class list created by ISMF.

Control Data Set Application: Use the CDS Class Application Selection panel to request that ISMF display CF cache structure names for all CF cache sets defined for a specific SCDS.

Changing the Size of Coupling Facility Cache Structures

DFSMS uses the initial structure size specified in the CFRM policy each time it connects to a CF cache structure. In each case, the ALTER(YES) keyword is specified, indicating to MVS that this structure can be dynamically reconfigured.

To alter the size of a CF cache structure, issue the SETXCF START command, using the following format:

```
SETXCF START,ALTER,STRNAME=CF_cache_structure_name,SIZE=newsize
```

where:

CF_cache_structure_name
: is the name of the cache structure being altered.

newsize
: is the new structure size in megabytes.

  This new size can be larger or smaller than the size of the current CF cache structure, but it cannot be larger than the maximum size specified in the CFRM policy.

MVS automatically starts the alter process in place, without disruptions to the application using the CF cache structure.

The alter function does not cross CF boundaries and does not take the place of the rebuild function. If you require a larger structure size than that specified in the CFRM policy, you must activate a new CFRM policy and rebuild the structure. The alter function changes the cache structure in place; you can use the rebuild function to move a cache structure to another CF.

See “Defining CF Cache Structures” on page 235 for information on estimating the size of CF cache structures.

Changing the Size of the Coupling Facility Lock Structure

You can rebuild the CF lock structure by redefining it to MVS. During the rebuild process, DFSMSdfp copies the currently held locks in the new CF lock structure.

The rebuild fails if the new CF lock structure is too small to contain the existing record table entries, or if the new lock structure cannot be defined with connectivity to all the systems sharing the current lock structure.

Changing IGDSMSxx Parameters to Support the Coupling Facility

The SYS1.PARMLIB IGDSMSxx member includes several parameters to support the CF:

CF_TIME

DEADLOCK_DETECTION

RLSINIT

RLS_MAX_POOL_SIZE

SMF_TIME

You can change the values for these parameters at any time during VSAM RLS processing. The new values are then used by all systems in the Parallel Sysplex, except for RLSINIT parameter values which are used only by the system accessing the changed parmlib member when SMSVSAM is next started.

To change values for these parameters, do one of the following:

- Issue the SETSMS operator command, specifying the parameter with different values.
- Issue the T SMS=*xx* command, where *xx* identifies an IGDSMSxx member where the parameter values are different than those currently in use.

The following list describes the parameters and their values:

{CF_TIME(*nnn*|3600)}
indicates the number of seconds between recording SMF type 42 records with subtypes 15, 16, 17, and 18 for the CF (both cache and lock structures). You can specify a value from 1 to 86399 (23 hours, 59 minutes, 59 seconds). The default is 3600 (one hour).

This keyword sets the interval time for the following SMF 42 subtypes:

SUBTYPE 15
CF storage class average response time

SUBTYPE 16
CF data set average response time

SUBTYPE 17
CF lock structure activity

SUBTYPE 18
CF cache partition summary

SUBTYPE 19
SMSVSAM least recently used statistics summary

{DEADLOCK_DETECTION(*iiii*|15,*kkkk*|4)}
specifies the deadlock detection intervals used by the Storage Management Locking Services.

iiii specifies the length in seconds of the local deadlock detection interval, as a one to four digit numeric value in the range 1-9999. The default is 15 seconds.

kkkk specifies the number of local deadlock cycles that must expire before global deadlock detection is run, as a one to four digit numeric value in the range 1-9999. The default is 4 cycles.

[RLSINIT({NO|YES})]
specify YES if you want the SMSVSAM address space started as part of system initialization or the V SMS,SMSVSAM,ACTIVE command. This value applies only to the system accessed by the parmlib member and is acted upon when SMSVSAM is next started. The default is NO.

[RLS_MAX_POOL_SIZE({nnnn|100})]
specifies the maximum size in megabytes of the SMSVSAM local buffer pool. SMSVSAM attempts to not exceed the buffer pool size you specify, although more storage might be temporarily used. Because SMSVSAM manages buffer pool space dynamically, this value does not set a static size for the buffer pool.

Use SMF 42, subtype 19 records to help you determine the maximum size of the SMSVSAM local buffer pool.

You can specify a two to four-digit numeric value, with 10 as the minimum value. If you specify a value less than 10, the field is set to 10. If you specify a value greater than 1500, SMSVSAM assumes there is no maximum limit. We recommend that you limit the size of the local buffer pool.

The default is **100MB**.

[SMF_TIME({YES|NO})]
specifies that the following SMF type 42 records are created at the SMF interval time, and that all of the indicated records are synchronized with SMF and RMF data intervals.

SUBTYPE 1
Buffer management statistics

SUBTYPE 2
Cache control unit statistics (IBM 3990 Storage Control Model 3)

SUBTYPE 15
Coupling facility storage class average response time

SUBTYPE 16
Coupling facility data set average response time

SUBTYPE 17
Coupling facility lock structure activity

SUBTYPE 18
Coupling facility cache partition summary

SUBTYPE 19
SMSVSAM least recently used statistics summary

This allows the customer to merge these SMF records for a specified time period and obtain both the 'system' view and the 'user' view of activity in the interval.

YES is the default. DFSMS creates the specified SMF record when the interval period expires and SMF sends the event notification signal.

If you specify YES, SMF_TIME overrides the following IGDSMSxx parameters: BMFTIME, CACHETIME, CF_TIME.

Changing the State of Coupling Facility Cache Structures and Volumes

You can use the VARY SMS command to control processing for volumes, data sets, or systems.

To alter the state of the specified CF cache structure, issue the VARY SMS command, using the following format:

`VARY SMS,CFCACHE(`*`CF_cache_structure_name`*`),ENABLE|QUIESCE`

When a CF cache structure is enabled, VSAM RLS data can be stored in the cache structure. This is the normal state of operations, and is the state the CF cache structure is in after the Parallel Sysplex has been IPLed. When a CF cache structure is quiesced, no VSAM RLS data can be stored in it.

To alter the state of the specified volume as it relates to all CF cache structures, issue the VARY SMS command using the following format:

`VARY SMS,CFVOL(`*`volser`*`),ENABLE|QUIESCE`

When a volume is CF-enabled, data contained on this volume can be stored in a CF cache structure. in the cache structure. This is the normal state of operations. When a volume is CF-quiesced, no data contained on it can be stored in a CF cache structure.

Use the VARY SMS CFVOL command if it is necessary to modify a volume without using the VSAM PUT/ERASE macros or DFSMSdss. This ensures that when the modified volume is again made available for VSAM RLS processing, CF cache structures do not contain downlevel data.

Note: Setting a volume to the CF-quiesced state does not stop SMS from selecting this volume during data set allocation. To stop SMS from selecting this volume, issue the VARY SMS command, using the following format: `VARY SMS,VOLUME(`*`volser`*`),DISABLE`

Selecting Data Sets for Coupling Facility Statistical Monitoring

You can use the VARY SMS command to specify which data set(s) are eligible for CF statistical monitoring. If statistical monitoring is on, SMF TYPE 42, subtype 16 records are produced. Use the following command format:

`VARY SMS,MONDS(`*`dsname`*`{`*`dsname`*`,...}),{ON|OFF}`

Select OFF to indicate that the specified data sets are no longer eligible for statistical monitoring. You can specify a full or partial data set name, with at least one high-level qualifier. An asterisk cannot be followed by other qualifiers. You can specify up to 16 data set specifications with each command. This command affects activity for the specified data sets across all systems in the Parallel Sysplex.

Recovering VSAM RLS Processing

This section describes:

- Recovering the CF lock structure
- Recovering a CF cache structure
- Recovering the SMSVSAM address space
- Recovering an SHCDS

Recovering the CF Lock Structure

At VSAM open time, the SMSVSAM address space checks to ensure that the CF lock structure is available. Record-level sharing cannot occur if the CF lock structure is unavailable or has failed.

A CF lock structure might fail and need to be rebuilt if the CF lock structure named IGWLOCK00 does not exist, or if it is not connected to the system attempting to open a VSAM data set for record-level sharing. In either case, DFSMS/MVS internally initiates a rebuild queueing all applications using VSAM RLS during the rebuild process. The rebuild is transparent to these applications.

If both the CF lock structure and the system fail, DFSMS/MVS cannot internally rebuild a new CF lock structure. In this case, all recoverable data sets which were open for VSAM RLS processing at the time of failure are converted to “lost locks”. They become unavailable to any processing besides recovery processing (such as backouts), and no new sharing is allowed until the recovery processing is complete. A new CF lock structure must be available to perform the recovery processing.

You might need to redefine or replace the CF lock structure, correct the problem causing its unavailability, or move the work requiring VSAM RLS to another system which has connectivity to an available CF lock structure.

Recovering a CF Cache Structure

In the event a CF cache structure fails, DFSMS attempts to rebuild it so that it remains available to the data with which it is associated. A CF cache structure is also rebuilt if DFSMS detects a loss of connectivity or an undersized cache structure.

If the rebuilding process is successful and connectivity resumes as before, all opens that were tied to the CF cache structure that failed are automatically reestablished. If the rebuild process fails, but another CF cache structure is defined in the cache set associated with the storage class and connectivity exists to that CF cache structure, then all opens are automatically reassigned to this alternate CF cache structure.

If the rebuild process fails and no alternate CF cache structure is available, any opens currently using the CF cache structure that failed are marked as broken. DFSMS fails the rebuilding process if the new CF cache structure is smaller than the failed structure, or if it does not have the same connectivity. In those cases where DFSMS cannot rebuild a CF cache structure, the next attempt to open a data set associated with the failed cache structure fails. You might need to redefine the cache structure or correct the connectivity problems.

Recovering the SMSVSAM Server Address Space

If the SMSVSAM server fails, it is automatically restarted. This is a complete re-initialization of the address space and the data space; all connections to prior instances of the SMSVSAM server are invalidated. Note that SMSVSAM is only started if the RLSINIT parameter of the IGDSMSxx parmlib member is set to YES.

The SMSVSAM server can be automatically restarted up to six times. If the limit is reached, the system issues message IGW418D. You respond to this message by indicating whether the automatic restart facility is to be re-enabled by the system or whether you want to manually enable it. If the automatic restart mechanism has been disabled, use the `VARY SMS,SMSVSAM,ACTIVE` command to restart SMSVSAM and reenable the automatic restart facility.

Recovering a Shared Control Data Set

You should always run with at least two active and one spare SHCDSs. If a permanent I/O error occurs for an active SHCDS, or if an SHCDS becomes inaccessible from one or more systems, it is automatically replaced by one of the spare SHCDSs. When a system is forced to run with only one SHCDS, it issues a message requesting that you add another active SHCDS and at least one spare SHCDS. If any system does not have access to an SHCDS, all opens for VSAM RLS processing are prevented on that system until an SHCDS becomes available.

The information in SHCDSs is continuously updated. Therefore, backup/restore procedures for SHCDSs are ineffective.

Use the following command formats for the VARY SMS command to add and delete SHCDSs:

To add a new, active SHCDS:	`VARY SMS,SHCDS(`*SHCDS_name*`),NEW`
To add a new, spare SHCDS:	`VARY SMS,SHCDS(`*SHCDS_name*`),NEWSPARE`
To delete either an active or a spare SHCDS:	`VARY SMS,SHCDS(`*SHCDS_name*`),DELETE`

Falling Back from VSAM RLS Processing

When you fall back from VSAM RLS processing, the following occurs: the SMSVSAM address space shuts down permanently on every system in the Parallel Sysplex; the SMSVSAM automatic restart capability is disabled; the lock structure IGWLOCK00 is deallocated, and all knowledge of SHCDSs and pending subsystem recovery is deleted.

The following sections describe some rules and considerations to follow when falling back from VSAM RLS processing. They also outline the fallback procedure to follow, providing you are only falling back from RLS processing and that you are staying at current product levels. If you are also falling back to different CICS, CICSVR, or DFSMS/MVS release levels, see "Understanding the Product Environment for VSAM RLS" on page 230 for a list of rules and considerations to follow.

Fallback Rules and Considerations

Consider and plan for the following situations before you decide to fall back from VSAM RLS processing:

- There might be outstanding recovery for VSAM RLS data sets. The fallback procedure results in the loss of locks protecting back out.
- RLS indicators in the catalog must be reset.
- There might be applications which can only function with VSAM RLS, and which cannot return to a VSAM NSR, LSR or global shared resources (GSR) environment.

Fallback Procedure

Note: This procedure is not intended to be used for normal or abnormal disabling of the SMSVSAM server.

Follow these steps to fall back from VSAM RLS processing:

1. Ensure that there are no outstanding recovery requirements.

   Use the access method services SHCDS LISTRECOVERY or LISTSUBSYSDS command to list all current recovery requirements known to SMSVSAM. Any recovery must be completed prior to continuing with fallback processing, or data integrity is compromised. Note that SMSVSAM is not aware of certain subsystem-related recovery, such as indoubt resolution.

   See *DFSMS/MVS Access Method Services for ICF* for a complete description of the access method services SHCDS command.
2. Activate an SMS configuration where all non-blank cache set specifications on storage classes are changed to blank cache set specifications.

   The SMS configuration should not include any cache sets defined in the base configuration. This ensures that the CF is only used for access to data sets which are already bound to a CF cache structure.
3. Ensure that you have no running applications which specify VSAM RLS processing, either specified in an ACB or using JCL.
4. Quiesce all CF cache structures.

   Use the VARY SMS command, as follows:

   ```
   VARY SMS,CFCACHE(CF_cache_structure_name),QUIESCE
   ```

   where:

   CF_cache_structure_name
   is the name of the cache structure being quiesced.

   Issue this command for each CF cache structure in your configuration. Use the D SMS,CFCACHE command to verify that all CF cache structures are quiesced.
5. Reset RLS indicators in all applicable catalogs, using the SHCDS CFRESET command.
6. When you delete RLS information in the catalog, revert to CICS FCT definitions, even if the CICS level remains unchanged

Be sure to complete all the preceding steps before continuing with the procedure to permanently shut down SMSVSAM and delete all knowledge of the lock structure and sharing control.

7. Change the value for the RLSINIT parameter in parmlib member IGDSMSxx to NO in *all* applicable parmlib members and activate the change.
8. Issue the following MVS command to disable the SMSVSAM server:

   ```
   *ROUTE ALL FORCE SMSVSAM,ARM
   ```

   Reply `C` to any outstanding IGW418D message. If FORCE SMSVSAM,ARM does not disable the SMSVSAM server, use FORCE SMSVSAM.
9. Issue the following command to complete the VSAM RLS fallback procedure:

   ```
   VARY SMS,SMSVSAM,FALLBACK
   ```

   This command issues message IGW523 to request confirmation. You should first ensure that all SMSVSAM servers are disabled, then respond with:

   ```
   FALLBACKSMSVSAMYES
   ```

   Any other response cancels the command.

   The FALLBACK command forces all lock table connections, and deletes the lock structure and the sharing control group, IGWXSGIS. If any of these steps fail, or if another FALLBACK command is already in process, the command is rejected.

Fallback is complete when message IGW524I is issued to the console that issued the FALLBACK command.

See “Understanding the Product Environment for VSAM RLS” on page 230 for a list of rules and considerations to follow if you are also falling back to different CICS, CICSVR, or DFSMS/MVS release levels.

Chapter 15. ACS Language Reference

This chapter provides general-use programming interface and associated guidance information. This chapter is intended to help you to write ACS routines.

ACS routines determine SMS classes and storage groups for all new data set allocations and for data set allocations that occur from converting, copying, recalling, restoring or moving data sets. For objects, ACS routines determine: storage group, when storing them; or storage class and management class, when storing or changing them, or during class transitions. You write ACS routines in the ACS language, which is a high-level programming language. You can write your ACS routines, one for each type of SMS class and one for your storage groups.

After writing the routines, you must translate them into an object form that SMS understands. A successful translation places the ACS object in a specified SCDS. After you activate the configuration contained in that SCDS, ACS routines govern storage management.

When you enter the ISMF Automatic Class Selection Application and select the EDIT option, you are linked to PDF Edit, where you can create or modify ACS routines. Leaving EDIT returns you to the ISMF Automatic Class Selection Application, where you can translate, validate, or test any ACS routine.

This chapter contains four main sections. The first section describes the ACS language constants. The second section describes *read-write variables*. The third section describes *read-only variables*, which the ACS routines use for comparison operations. The fourth section describes the ACS language statements and illustrates the use of the statements in a storage group selection routine.

Constants

You can use four types of constants in ACS routines:

Numeric
A numeric is a string containing up to ten characters, 0 through 9. You can use numerics in comparison operations involving the &NQUAL, &NVOL, and &RETPD read-only variables, which are discussed in “Read-Only Variables” on page 261.

KB, MB
KB and MB are suffixes for numeric constants, such as 200KB and 10MB. One KB = 1,024 bytes while one MB = 1,048,576 bytes. Any comparison operation involving the &SIZE and &MAXSIZE read-only variables require that you use KB or MB. They are discussed in “Read-Only Variables” on page 261. The maximum prefix value for KB is 2147483647. The maximum prefix value for MB is 2097151.

Literal A literal is a character string, such as 'SYS1.PARMLIB', that is enclosed in single quotation marks. The maximum length of a literal is 255 characters. If you want a literal to contain a single quotation mark, such as PAYROLL'SDATA, then you must specify two single quotation marks: 'PAYROLL''SDATA'.

Mask A mask is a character string, such as SYS1.*LIB, that is *not* enclosed in single quotation marks. You can use a mask to represent job names, volume serial numbers, or other system values that have a common string of characters, such as all volume serial numbers that begin with IMS. You can also use a mask to represent data set, object or collection names that have a common string of characters.

A mask must begin with an alphabetic character, numeric character (0-9), national character ($, @, #), asterisk (*), or percent sign (%). The three characters “*”, “%”, and “.” have special significance in a mask. In addition, the characters ″-″ and ″+″ cannot be used in masks. These characters are reserved for use as continuation characters. The following sections describe the rules for using both the simpler masks and the slightly more involved data set masks. See “FILTLIST Statement” on page 278 for an explanation of the use of the mask characters.

Simple Mask Rules

The following rules apply to the special characters in a simple mask containing a single level name:

- An asterisk, “*”, means that zero or more characters can be present in its place.
- Two or more adjacent asterisks are not allowed within a simple mask.
- A “%” represents exactly one non-blank character. “%%%” represents three character positions.

Simple Mask Examples

TSO* All names of any length beginning “TSO”

XYZ All names of any length having three adjacent characters “XYZ”

IMS%%%
All six-character names beginning “IMS”

***%WK%%**
All names where the second and third characters of the last five (or only five) are “WK”

Data Set Mask Rules

The following rules apply to the special characters in a data set mask:

- You can separate data set qualifiers with periods, “.”.
- Each qualifier has a maximum length of eight characters. The maximum length for the entire data set mask is 44 characters.
- A “%” represents a single character position. “%%%” represents three character positions.
- A single “*” by itself indicates that at least one qualifier is needed to occupy that position. A “*” within a qualifier means that zero or more characters can be present.
- A qualifier can be a single “*”
- A “**” means that zero or more qualifiers can be present.
- A “**” cannot appear with any other characters within a qualifier.
- Three or more adjacent “*” are not allowed within a qualifier.
- The read-only variables which cannot be used for comparisons are: &ACSENVIR, &DSNTYPE, &DSORG, &DSTYPE, &LABEL, &RECORG, and &XMODE.

Data Set Mask Examples

SYS1.**
All names where the first (or only) qualifier is “SYS1”

****.OUTLIST**
All names where the last (or only) qualifier is “OUTLIST”

.*.PAYROLL.*.SALARY.
All names with six qualifiers where the third qualifier is “PAYROLL” and the fifth qualifier is “SALARY”

***.%%TEST.*.DATA**
All names with four qualifiers where the second qualifier has six characters ending in “TEST” and the fourth qualifier is “DATA”

****.*ABC*.****
All names where some (or only) qualifier contains the characters “ABC” (or only “ABC”)

LABMGR..DATA**
All names where the first qualifier is “LABMGR” and the last qualifier is “DATA”

Read-Write Variables

You write ACS routines to assign values to read-write variables. You can also use read-write variables as values in comparison operations. These read-write variables *are* case sensitive. The ACS language has four read-write variables:

&DATACLAS
&STORCLAS
&MGMTCLAS
&STORGRP

The &STORGRP read-write variable should only be used in the storage group routine. It is null on input to the routine unless VOL=REF is specified.

If the read-write variables are explicitly specified by the user, they have an initial value that might be overridden by the ACS routine. If the value is not overridden and the initial name is not defined in a currently active configuration, the allocation fails.

You must specify a read-write variable on the PROC statement of the corresponding ACS routine. See “PROC Statement” on page 277 for details.

Each ACS routine can set only its corresponding read-write variable:

- The data class routine can set only &DATACLAS.
- The storage class routine can set only &STORCLAS.
- The management class routine can set only &MGMTCLAS.
- The storage group routine can set only &STORGRP.

The ACS routines assign values to read-write variables using the SET command, which is explained in “SET Statement” on page 279.

In an ACS routine, you can assign an alphanumeric name enclosed in single quotation marks to the read-write variables. Also, you can assign a list of up to fifteen alphanumeric storage group names, each enclosed in single quotation marks, to the &STORGRP read-write variable. If more than one storage group name exists in the list, then each name must be enclosed in single quotes and separated by commas (for example, 'SG1', 'SG2').

Table 18 indicates which read-write variables you can set, which ones you can use for comparisons, and which ones are invalid, for each of the ACS routines:

Table 18. Using Read-Write Variables in ACS Routines

	Read-Write Variable			
ACS Routine	**&STORGRP**	**&MGMTCLAS**	**&STORCLAS**	**&DATACLAS**
Storage group	Set/Compare	Compare	Compare	Compare
Management class	Invalid	Set/Compare	Compare	Compare
Storage class	Invalid	Compare	Set/Compare	Compare
Data class	Invalid	Compare	Compare	Set/Compare

Using Read-Write Variables with Volume Reference (VOL=REF)

When volume reference (VOL=REF) is used, the storage group of the referenced data set is passed to the ACS routines in the &STORGRP read-write variable. However, keep in mind the following:

- The storage group name might not be available if the reference is to a data set on SMS-managed tape. This is because private tapes can be entered into a tape library with a blank storage group name. In this case, the AS routine should use the &LIBNAME read-only variable to determine the storage group for the referenced data set. In this case, the referenced and referencing data sets must reside in the same storage group.
- If the reference is to a new data set, there can be multiple candidate storage groups for the referenced data set and the actual storage group might not have been selected yet. In this case, only the first candidate storage group is passed as input to the ACS routines, and this might not be the storage group in which the referenced data set is eventually allocated.

Using Read-Write Variables with Data Set Stacking

When a data set stacking inconsistency is detected, the ACS routines are re-invoked. When they are available, the following values are passed as input to the ACS routines:

- The storage group of the primary data set
- The storage class of the primary data set
- The management class selected by the previous invocation of the management class ACS routine for the stacked data set
- The data class selected by the previous invocation of the data class ACS routine for the stacked data set

Values might not be available for the following reasons:

- No storage class and storage group is available from the primary data set if it is directed to non-SMS-managed media.
- No management class is available for the stacked data set if it was initially assigned no storage class.

- If SMS was invoked by JES3, it is unable to access the work areas to obtain the storage class, management class, or data class.
- The storage group name might not be available if the primary data set is on SMS-managed tape. This is because private tapes can be entered into a tape library with a blank storage group name. In this case, the ACS routine should use the &LIBNAME read-only variable to determine the storage group for the stacked data set. In this case, the primary data set and the stacked data set must reside in the same storage group.
- If the primary data set is new, there might be multiple candidate storage groups for it and the actual storage group might not have been selected yet. In this case, only the first candidate storage group is passed as input to the ACS routines, and this might not be the storage group in which the primary data set is eventually allocated.

Read-Only Variables

Most ACS variables are read-only. Read-only variables contain data set and system information, and they reflect what is known at the time of the allocation request. You can use read-only variables in comparison operations, but you cannot change their values.

All of the read-only variables appear below in Table 19. The read-only variables *are* case sensitive. The following pages explain the uses of the read-only variables.

Note: In the data class ACS routine, the &DSNTYPE, &DSORG, &MAXSIZE, &NVOL, &RECORG, and &SIZE variables all default to null if no corresponding value is specified in the JCL.

Table 19. Read-Only Variables

&ACCT_JOB	&DEF_STORCLAS	&HLQ	&MEMNQUAL	&RECORG
&ACCT_STEP	&DSN	&JOB	&MSDEST	&RETPD
&ACSENVIR	&DSNTYPE	&LABEL	&MSPARM	&SIZE
&ALLVOL	&DSORG	&LIBNAME	&MSPOLICY	&SYSNAME
&ANYVOL	&DSOWNER	&LLQ	&MSPOOOL	&SYSPLEX
&APPLIC	&DSTYPE	&MAXSIZE	&MSVGP	&UNIT
&DD	&EXPDT	&MEMHLQ	&NQUAL	&USER
&DEF_DATACLAS	&FILENUM	&MEMLLQ	&NVOL	&XMODE
&DEF_MGMTCLAS	&GROUP	&MEMN	&PGM	

Name Description

&ACCT_JOB
The accounting information from the JOB statement. (For a description of the indexing function for accounting information, see “Special Functions” on page 274.)

Type: Literal

Max value: 142 characters

&ACCT_STEP
The accounting information from the EXEC statement. This information is refreshed for each step in the job. (For a description of the indexing function for accounting information, see “Special Functions” on page 274.)

Type: Literal

Max value: 142 characters

&ACSENVIR

The environment in which the ACS routine was invoked, one of:

RECALL
for data set recall operations

RECOVER
for data set recover operations

RENAME
for data set alter rename operations

CONVERT
for data set convert in place operations

ALLOC
for new data set allocations (this is the default)

STORE
OSREQ object store environment

CHANGE
OSREQ object change environment

CTRANS
OSMC object class transition environment

other installation exit can set its own value before re-invoking ACS

Type: Literal

Max value: 8 characters

&ALLVOL

The volume serial numbers specified for data set allocation when &ACSENVIR is not recall or recover. When the environment is recall or recover, &ALLVOL is either the volume serial number on which the data set resided at the time it was migrated or backed up, or the volume serial number specified as the target volume of the recall or recover. See “Special Functions” on page 274 for usage information.

Type: Literal

Max value: 6 characters

The &ALLVOL ACS read-only variable contains the following values when you use VOL=REF:

- ’REF=SD’ (the volume reference is to an SMS-managed DASD or VIO data set)
- ’REF=ST’ (the volume reference is to an SMS-manged tape data set)
- ’REF=NS’ (the volume reference is to a non-SMS-managed data set)

&ANYVOL

The volume serial numbers explicitly specified for the volume(s) if &ACSENVIR is not recall or recover. When the environment is recall or recover, &ANYVOL is either the serial number on which the data set resided at the time it was migrated or backed up, or the volume serial number specified as the target volume of the recall or recover. See “Special Functions” on page 274 for usage information.

Type: Literal

Max value: 6 characters

The &ANYVOL ACS read-only variable contains the following values when you use VOL=REF:

- 'REF=SD' (the volume is reference to an SMS-managed DASD data set)
- 'REF=ST' (the volume is reference to an SMS-manged tape data set)
- 'REF=NS' (the volume is reference to a Non-SMS-managed data set)

&APPLIC
The name of the application associated with the resource owner of the data set (which is set only if RACF is installed and ACSDEFAULTS is YES in IGDSMSxx).

Type: Literal

Max value: 8 characters

&DD DDNAME in the DD statement of the data set.

Type: Literal Max value: 8 characters

&DEF_DATACLAS
The data class name associated with the resource owner of the data set (set only if RACF is installed and ACSDEFAULTS is YES in IGDSMSxx).

Type: Literal Max value: 8 characters

&DEF_MGMTCLAS
The management class name associated with the resource owner of the data set (set only if RACF is installed and ACSDEFAULTS is YES in IGDSMSxx).

Type: Literal

Max value: 8 characters

&DEF_STORCLAS
The storage class name associated with the resource owner of the data set (set only if RACF is installed and ACSDEFAULTS is YES in IGDSMSxx).

Type: Literal

Max value: 8 characters

&DSN The name of the data set or collection for which ACS processing is taking place. For VSAM data sets, only the cluster name is passed to the ACS routine; the component name(s) are not.

If the data set has an absolute or relative generation number, it is stripped from &DSN The generation number is the low-level qualifier of the data set name.

Note: See *OS/390 MVS JCL Reference* for the data set naming rules.

Type: Literal

Max value: 44 characters

&DSNTYPE
The data set name type, one of:

EXC Extended format data set is preferred. The data set allocation is attempted in non-extended format if the necessary system resources for extended are not available.

EXR Extended format data set is required. The data set allocation fails if unable to allocate in extended format.

HFS Hierarchical file system data set

LIBRARY
PDSE in SMS-managed storage; partitioned data set in non-managed storage

PDS Partitioned data set

null No value specified

Type: Literal

Max value: 7 characters

&DSORG
The data set organization, one of:

PS Physical sequential

PO Partitioned

VS VSAM organization

DA BDAM organization

null No value specified

Type: Literal

Max value: 2 characters

&DSOWNER
The name of the user or group that owns the data set (set only if RACF is installed).

Type: Literal

Max value: 8 characters

&DSTYPE
The data set type, one of:

GDS One generation data set of a generation data group, or any data set allocated with a relative generation number (such as A.B.C(+1)) or an absolute generation number (such as A.B.C.G0000V00).

PERM Standard permanent data sets

TEMP Temporary data sets

null None of the above.

Type: Literal

Max value: 8 characters

&EXPDT
The expiration date in the form YYYYDDD where YYYY is a year between 1900 and 2050 and DDD is a day of year between 1 and 366.

Type: Literal

Max value: 7 characters

&FILENUM
The value of the FILENUM ACS read-only variable. This variable corresponds to the data set sequence number on the JCL LABEL parameter. The default is 1. This field is optional.

Type: Numeric

Max value: 4 characters

&GROUP
The RACF-defined group to which you are connected, or the group specified in the GROUP keyword on the JCL JOB statement. If the environment is recall or recover, &GROUP is set only if the requester of the recall or recover is not a DFSMShsm authorized user. When DFSMShsm invokes the ACS routines, &GROUP is the group associated with &USER

Type: Literal

Max value: 8 characters

&HLQ The high-level (first) qualifier of the data set or collection name.

Type: Literal

Max value: 8 characters

&JOB The job name, the started task name, or the TSO/E userid from the JOB statement, depending on the execution mode (&XMODE). (See "Determining Distributed FileManager/MVS Data Set Creation Requests" on page 275 for Distributed FileManager/MVS usage information.)

Type: Literal

Max value: 8 characters

&LABEL
The value of the LABEL ACS read-only variable. This variable corresponds to the label field of the JCL LABEL parameter. Allowable values are NL, AL, SL, NSL, SUL, AUL, BLP, LTM or blank. The default is IBM Standard Label. This field is optional.

Type: Literal

Max value: 3 characters

&LIBNAME
The name for the LIBNAME ACS read-only variable, can contain a 1 to 8 character tape library name. This field is optional.

Type: Literal

Max value: 8 characters

&LLQ The low-level (last) qualifier of the data set or collection name.

Type: Literal

Max value: 8 characters

&MAXSIZE
The maximum size (in KB or MB) of a new data set. For non-VSAM data sets, the value is &SIZE plus 15 extents. For VSAM data sets, the value is &SIZE plus 122 extents. See "Using Read-Only Variables" on page 269 for more information about the values of &MAXSIZE and &SIZE for VSAM data sets. Also see "Constraints When Using Read-Only Variables" on page 271.

| Type: Suffixed numeric

Max value: 2147483647 for KB, 2097151 for MB

&MEMHLQ
The high-level (first) qualifier of the object name.

Type: Literal

Max value: 8 characters

&MEMLLQ
The low-level (last) qualifier of the object name.

Type: Literal

Max value: 8 characters

&MEMN
The name of an object.

Type: Literal

Max value: 44 characters

&MEMNQUAL
The number of qualifiers in the object name.

Type: Numeric

Max value: 22

&MSDEST
The destination, specified in data set name format, for a tape management system-driven tape allocation. This value is specified through an external exit[5]. The data set name format lets you specify a sequence of destinations to be identified, where each qualifier is a specific destination. For example, a data set vaulted first at location OUTD and then sent to OLTS could have an MSDEST of 'OUTD.OLTS'. The actual values depend on the support provided by your tape management system.

Type: Alphanumeric

Max value: 44 characters

&MSPARM
Additional information related to a tape management system-driven tape allocation. This is a variable length field. The value is specified through an external exit[6].

Type: Alphanumeric

Max value: 256 bytes, including a 2–byte length field for each value specified

&MSPOLICY
The name of a management policy associated with data in a Virtual Tape Server (VTS), for a tape management system-driven tape allocation. This value is specified through an external exit[7].

Type: Alphanumeric

5. Import/Export support is available with APAR OW36342 or OW36343. If you are planning on using the Pre-ACS routine exit routine, APAR OW36351 or OW36352 should be installed.

6. Import/Export support is available with APAR OW36342 or OW36343. If you are planning on using the Pre-ACS routine exit routine, APAR OW36351 or OW36352 should be installed.

7. Import/Export support is available with APAR OW36342 or OW36343. If you are planning on using the Pre-ACS routine exit routine, APAR OW36351 or OW36352 should be installed.

Max value: 8 characters

&MSPOOL
A tape pool name associated with the data set being allocated. In a system-managed tape environment with scratch pool support, this variable might be used to specify a default storage group, where the tape storage group is equivalent to the tape pool specified in the variable. This value is specified through an external exit[8].

Type: Alphanumeric

Max value: 8 characters

&MSVGP
The group name for mass storage subsystem (MSS) volumes.

Type: Literal

Max value: 8 characters

Note: MVS no longer supports MSS. This variable has no effect.

&NQUAL
The number of qualifiers in the data set or collection name.

Type: Numeric

Max value: 22

&NVOL
The maximum of the volume count, UNIT count, and number of explicit VOL=SER specifications.

Type: Numeric

Max value: 2147483647

&PGM The name of the program the system is running. (See "Determining Distributed FileManager/MVS Data Set Creation Requests" on page 275 for Distributed FileManager/MVS usage information.)

Type: Literal

Max value: 8 characters

&RECORG
The data set record organization, one of:

KS VSAM key sequenced (KSDS)

ES VSAM entry sequenced (ESDS)

RR VSAM relative record (RRDS)

LS VSAM linear

null No value specified

Type: Literal

Max value: 2 characters

&RETPD
The retention period (nnnn days).

8. Import/Export support is available with APAR OW36342 or OW36343. If you are planning on using the Pre-ACS routine exit routine, APAR OW36351 or OW36352 should be installed.

Type: Numeric

Max value: 2147483647

&SIZE The primary amount of space (in KB or MB) requested for a new data set or the amount of space actually used in an existing data set on a DASD volume. See “Using Read-Only Variables” on page 269 for more information about the values of &MAXSIZE and &SIZE for VSAM data sets.

Type: Suffixed numeric

Max value: 2147483647 for KB; 2097151 for MB

&SYSNAME
Specifies the system name of the system on which the ACS routine is executing. This field is optional. See “Using Read-Only Variables” on page 269 for usage information.

Type: Literal

Max value: 8 characters

&SYSPLEX
Specifies the Parallel Sysplex name of the system on which the ACS routine is executing. This field is optional. See “Using Read-Only Variables” on page 269 for usage information.

Type: Literal

Max value: 8 characters

&UNIT
IBM-supplied or installation defined generic name for a device type (for example 3380, SYSDA). For additional possible settings for the &UNIT variable, see “ACS Routine Environments” on page 160.

Type: Literal

Max value: 8 characters

Note: A slash (/) preceding a four digit number represents a unit address (i.e. /3380).

Note: When you allocate a tape data set with DISP=MOD and no unit information specified in the JCL, this variable is blank and SMS might attempt to manage the tape data set as a DASD-resident data set.

&USER
The user ID of the person allocating the data set. When DFSMShsm invokes the ACS routines, &USER is either the requestor of the recall or recover, or the user ID of the DFSMShsm address space. If the environment is recall or recover, the variable is set only if the requestor of the recall or recover is not a DFSMShsm authorized user. (See “Determining Distributed FileManager/MVS Data Set Creation Requests” on page 275 for Distributed FileManager/MVS usage information.)

Type: Literal

Max value: 8 characters

&XMODE
The execution mode in which the data set is being allocated, one of:

BATCH
Batch execution mode

TSO TSO execution mode

TASK A started address space

Type: Literal

Max value: 8 characters

Using Read-Only Variables

The following sections describe various considerations for using read-only variables:

Initializing Read-Only Variables

SMS derives the values of read-only variables before it invokes the ACS routines. The values are based on what is known at the time of the allocation request (for example, if a unit name has been specified on the allocation request, then the &UNIT variable contains the specification).

Note: Note that &DSNTYPE is not initialized with the DSNTYPE default specified in the IGDSMSxx member in SYS1.PARMLIB.

Using Default Read-Only Variables

Read-only variables are defaulted from data class (if assigned) before storage class, management class, and storage group ACS routines are invoked.

Ensuring Correct Values for &SIZE and &MAXSIZE

If you allocate a data set using the TSO/E ALLOCATE command and you do not explicitly specify space requirements, then &SIZE and &MAXSIZE do not contain the correct values. Instead they both contain a value of zero. If your ACS routines rely on the values of &SIZE or &MAXSIZE in this situation, the data set might be assigned to the wrong class or group.

For a VSAM data set definition, the &SIZE and &MAXSIZE read-only variables reflect the space value specified in the CLUSTER component. If one is not specified in the CLUSTER component, then the space value specified in the DATA component is used. If a space value also is specified for the INDEX component *and* it is of the same type of space unit; for example, both are in tracks, cylinders, KB or MB, it is added to what was specified for the DATA component. If the INDEX component space unit is not of the same type as specified for the DATA component, it is ignored and not added to &SIZE or &MAXSIZE.

For DFSORT work data sets, &SIZE and &MAXSIZE are zero. In the RECALL environment, &SIZE and &MAXSIZE are zero for empty partitioned data set and PS data sets because the size of the existing data set is not known at the time that DFSMS/MVS runs its ACS routines.

Using OAM Read-Only Variables

&MEMN, &MEMHLQ, &MEMLLQ and &MEMNQUAL are used to name an object in a collection. These read-only variables are valid only when &ACSENVIR is equal to one of the three valid OAM environments (STORE, CTRANS, CHANGE), otherwise, the passed value is nullified by ACS before invoking the ACS routines.

Using &DSN for a Partitioned Data Set

For a partitioned data set, &DSN consists only of the dataset name. A member name, if specified, is not be part of the value of the &DSN or &MEMN variables. There is no class selection capability based on member name.

When the Value of &USER Is Null

Not all read-only variable values are significant during the actual operation of any ACS routine. For example, the value of &USER could be null in the storage group selection routine if no user ID had been specified on the JOB statement or determined from the environment.

The user ID (and group ID) is only available if specified using JCL or if RACF (or any other security product) is active. For example, on a RESTORE, if a user ID is specified, the ACERO passed by DFSMSdss does not contain the user ID, and because RACF is not active, DFSMS/MVS does not interrogate the ACEE to fill it in.

Using the &SYSNAME and &SYSPLEX Read-Only Variables

Due to connectivity constraints on old DASD and tape, it might be necessary to know the system and Parallel Sysplex where an ACS routine is being executed in order to direct the allocation to a storage group(s) which is accessible from the current system. To support this function, SMS ACS processing uses the &SYSNAME and &SYSPLEX read-only variables.

The &SYSPLEX variable should not be used in ACS routines for JES3 systems as Parallel Sysplex name support is not supported in a JES3 environment. In addition, ACS routines for JES3 systems should not rely too heavily on the &SYSNAME variable, as the system on which the ACS routines are run (during converter/interpreter), might have nothing to do with the system on which the job is eventually executed.

The &SYSNAME and &SYSPLEX variables are not supported on systems running releases prior to DFSMS/MVS 1.2.

Read-Only Variables on a Data Set Rename

On a data set rename using &ENVIR=RENAME, not all read-only variables are passed to the ACS routines. The following read-only variables *are* passed to the ACS routines:

- &ACSENVIR
- &APPLIC (set by ACS routines)
- &DEF_DATACLAS (set to null by ACS routines)
- &DEF_MGMTCLAS (set to null by ACS routines)
- &DEF_STORCLAS (set to null by ACS routines)
- &DSOWNER
- &DSN
- &DSORG
- &DSTYPE
- &EXPDT
- &GROUP
- &SYSNAME (set by ACS routines)

- &SYSPLEX (set by ACS routines)
- &USER

The following read/write variables are also passed to the ACS routines:

- &DATACLAS
- &MGMTCLAS
- &STORCLAS

WRITE statements are not passed to the ACS routines after a data set rename.

Constraints When Using Read-Only Variables

The following sections summarize some constraints when using read-only variables.

Read-Only Variables Not Allowed in Storage Group Routine

The following read-only variables are *not* allowed in the storage group selection routine:

&ACCT_JOB
&ACCT_STEP
&DD
&JOB
&MSVGP
&PGM
&XMODE

Read-Only Variables in Different Environments

The following read-only variables are *not* passed to any ACS routine unless the environment is specified as &ACSENVIR=ALLOC:

&ACCT_JOB
&ACCT_STEP
&DD
&GROUP
&JOB
&MSGVP
&PGM
&USER
&XMODE

Notes:

1. If the environment is a non-IBM supported environment, the following read-only variables *are* passed regardless of what is specified in &ACSENVIR:

   &JOB
   &DD
   &PGM
2. If the environment specified in &ACSENVIR is RECALL, RENAME, or RECOVER, the following read-only variables *are* passed:

   &GROUP
   &USER

Read-Only Variables Not Available when LIKE is Used

When you use the LIKE parameter on a JCL DD statement or the ALLOCATE command, the following read-only variable values are not available to the ACS routines:

&DSNTYPE

&MAXSIZE

&RECORG

&SIZE

Thus, a data set allocated like a second data set might go into in a different storage group than the second data set.

OAM Read-Only Variables

The following read-only variables are passed to all but the data class selection routine when &ACSENVIR is STORE, CTRANS or CHANGE:

&MEMN

&MEMHLQ

&MEMLLQ

&MEMNQUAL

Comparison Operators

Comparison operators allow you to determine the relationship between two values. The following comparison operators are allowed:

This: Means this:

GT or >
Greater than

LT or <
Less than

NG or ¬>
Not greater than

NL or ¬<
Not less than

EQ or =
Equal

NE or ¬=
Not equal

GE or >=
Greater than or equal

LE or <=
Less than or equal

Alphabetic characters are sorted before digits (A-Z come before 0-9). The following comparison is true for all high-level qualifiers alphanumerically greater than “M”:

```
IF &HLQ > 'M' THEN . . .
```

For FILTLIST or mask comparisons, only EQ and NE are valid. See “FILTLIST Statement” on page 278 for details.

Comparison Rules

The following rules apply to comparisons:

1. For a comparison to be valid, one operand must either be a read-only variable or a read-write variable and the other operand must be a constant (any of the four types), a read-only variable, or a FILTLIST name.
2. Numerics are right justified.
3. Literals are left justified and padded with blanks.
4. Type checking is done to ensure that numeric read-only variables are not being compared to characters (literals) and that character (literal) read-only variables are not being compared to numbers. &NQUAL, &NVOL, &SIZE, &MAXSIZE &MEMNQUAL and &RETPD are the only numeric read-only variables.
5. Limited length checking of read-only variables with their maximum length values is performed to ensure that the maximum lengths are not exceeded. For example, the literal to which &DSN is being compared must be no longer than 44 characters.

   See “Read-Only Variables” on page 261 for maximum lengths.

Boolean Expressions

You can use the following Boolean operators in any ACS routine:

This: **Means this:**

AND or &&;
: And

OR or |
: Or

Expressions in parentheses are processed first. In the following example, the set of OR expressions are processed first, so that

```
WHEN ((CONDITION 1) OR
      (CONDITION 2) OR
      (CONDITION 3) AND
      (CONDITION 4)) SET ...
```

is processed as follows:

```
WHEN
 (CONDITION 1 AND CONDITION 4) OR
 (CONDITION 2 AND CONDITION 4) OR
 (CONDITION 3 AND CONDITION 4) SET ...
```

If you want the AND expression be processed before OR, the AND expression must be included in a set of parentheses. In the following example, AND is processed first so that

```
WHEN ((CONDITION 1) OR
       (CONDITION 2) OR
       ((CONDITION 3) AND
       (CONDITION 4))) SET...
```

is processed as follows:

```
WHEN ((CONDITION 1) OR
      (CONDITION 2) OR
      (CONDITION 3 AND CONDITION 4)) SET...
```

Special Functions

The ACS language provides the following three indexing functions, which allow you to make class selections based on specific details about each data set.

Table 20. Indexing Functions for Read-Only Variables

Selection Routine	Data Set Qualifier	&ALLVOL &ANYVOL	Accounting Information
Data class	Yes	Yes	Yes
Storage class	Yes	Yes	Yes
Management class	Yes	Yes	Yes
Storage group	Yes	Yes	No

Data Set Qualifier Indexing Function

The data set qualifier indexing function lets you index the &DSN and &MEMN variables (for accessing particular qualifiers):

&DSN(1)
first qualifier of the data set name

&DSN(3)
third qualifier of the data set name

&DSN(&NQUAL)
last qualifier of the data set name

&MEMN(1)
first qualifier of the object name

&MEMN(2)
second qualifier of the object name

&MEMN(&MEMNQUAL)
last qualifier of the object name

The only accepted values for indexes are numbers (1 through 22) and the read-only variables, &NQUAL and &MEMNQUAL.

&ALLVOL and &ANYVOL Functions

These functions let you compare the volume serial number(s) explicitly specified on input with a comparison variable (for example, a FILTLIST variable). The use of &ALLVOL in a comparison expression returns a true value if ALL of the input volsers satisfy the desired condition. The use of &ANYVOL returns a true value if ANY of the input volsers satisfies the desired condition. For example, let IMS101, IMS102, and TSO191 be the input volsers to the routine in Figure 113 on page 275:

```
PROC STORCLAS

    IF &ALLVOL = IMS* THEN
       (code left out)
    SELECT
      WHEN (&ANYVOL = TSO*)
       (code left out)
    END
END
```

Figure 113. Example of Constraints when Using the &ALLVOL and &ANYVOL Read-Only Variables

The IF statement is false because not *all* volumes match. While IMS101 and IMS102 satisfy the IMS* mask, TSO191 does not.

The WHEN statement is true, because *any* (at least one) volume does match. While IMS101 and IMS102 fail to satisfy the TSO mask, TSO191 does satisfy the mask.

Note: For a detailed explanation of the IF statement, see “IF Statement” on page 280.

Accounting Information Indexing Function

Allows you to reference specific fields in the JOB or STEP account information. A field is defined as a unit of data which is separated by comma(s) in the account information. The use of &ACCT_JOB(n) and &ACCT_STEP(n), where n is the field number, indicates that indexing is requested. If either &ACCT_JOB or &ACCT_STEP is specified without an index, then the default action is to access the first field of the accounting information (for example, &ACCT_JOB(1) or &ACCT_STEP(1) result by default) .

```
&ACCT_JOB(1)       first field of the JOB accounting information
&ACCT_JOB          first field of the JOB accounting information (default)
&ACCT_JOB(3)       third field of the JOB accounting information
&ACCT_STEP         first field of the STEP accounting information (default)
&ACCT_STEP(5)      fifth field of the STEP accounting information
```

The only accepted values for indexes are numbers from 1 to 71.

Determining Distributed FileManager/MVS Data Set Creation Requests

The ACS routines are used to determine the SMS classes for data sets created by Distributed FileManager/MVS. You can use the &JOB, &PGM, and &USER read only variables to distinguish Distributed FileManager/MVS data set creation requests. The following examples show how each of these variables can be used.

&JOB

The value of the &JOB variable is the job specified in the Advanced Program to Program Communications (APPC) transaction program established by the installation systems programmer. Figure 114 on page 276 shows how each &JOB can be used to determine the SMS storage class for a data set being created using Distributed FileManager/MVS:

```
PROC &STORCLASS
 .
 .
 IF &JOB =  'GDEDFM' AND &STORCLAS = '' THEN
   SET &STORCLAS = 'DFMCLASS'
 ELSE
   .
   .
   .
```

Figure 114. Example Showing &JOB Read-Only Variable

&PGM

The value of the &PGM variable is specified in the transaction program profile and is always GDEISASB. Figure 115 shows how &PGM can be used to determine the SMS storage class for a data set being created using Distributed FileManager/MVS:

```
PROC &STORCLAS
 .
 .
 IF &PGM = 'GDEISASB' AND &STORCLAS = '' THEN
   SET &STORCLAS = 'DFMCLASS'
 ELSE
   .
   .
   .
```

Figure 115. Example Showing &PGM Read-Only Variable

&USER

You can designate a specific user ID under which all Distributed FileManager/MVS transaction programs are to run. You can then check the user ID to determine if the data set is being created using Distributed FileManager/MVS. The user ID is specified in the job statement in the Distributed FileManager/MVS transaction program profile.

Figure 116 shows how &USER can be used to determine the SMS storage class for a data set being created using Distributed FileManager/MVS:

```
PROC &STORCLAS
 .
 .
 IF &USER = 'user_ID' AND &STORCLAS = '' THEN
   SET &STORCLAS = 'DFMCLASS'
 ELSE
   .
   .
   .
```

Figure 116. Example Showing &USER Read-Only Variable

Statements

This section describes the function and syntax of the ACS language statements that you can use when writing ACS routines.

The continuation characters “+” and “-” allow you to extend literal constants to the next line. To ignore the leading blanks on the following line, use “+”. If you want to include the leading blanks on the next line as part of a literal, then use a “-”. You cannot continue masks, numbers, KB or MB numerics, or keywords.

The maximum number of nesting levels for any combination of ACS statement types is thirty-two. (For example, a nested IF statement is one that appears within an IF statement.)

Comments begin with a slash-asterisk pair, “/*”, and end with an asterisk-slash pair, “*/”.
- You can place comments anywhere within an ACS routine where a delimiter might appear.
- Comments cannot be nested; each comment ends at the first occurrence of an asterisk-slash pair, “*/”.
- Asterisks within the comment statement are treated as a special character. A maximum of 500 asterisks can be included in a single comment statement (multiple comment lines not ended with “*/”). This could affect the number of lines allowed in comment continuation.

The statement types are defined as follows:

Statement
: **Description**

PROC
: Start of an ACS routine

FILTLIST
: Definition of filter criteria

SET
: Assigns a value to a read-write variable

DO
: Start of statement group

IF
: Provides conditional statement execution

SELECT
: Defines a set of conditional execution statements

EXIT
: Causes immediate termination of the ACS routine and can be used to force allocation failures

WRITE
: Sends a message to the end user

END
: End of statement group (DO or SELECT) or ACS routine (PROC).

PROC Statement

PROC is the first statement of each ACS routine. It identifies the ACS routine and which read-write variable the routine sets. You can precede the PROC statement with blank lines or comments, but not with other statements. For TSO CLIST compatibility, you can place a blank and then a number, such as 0 or 1, after the PROC statement. The number does not affect ACS language processing. To identify an ACS routine and the value it is to determine, you must specify a read-write variable at the end of the PROC statement. You must also place an END statement at the end of each ACS routine.

PROC *<n> read-write variable*
- *n* is optional and can contain any numeric value.

- *read-write variable* is a mandatory value that can be DATACLAS, STORCLAS, MGMTCLAS, or STORGRP. You can optionally precede the variable with an ampersand, &.

```
PROC 1 DATACLAS
PROC 0 &STORCLAS
PROC &MGMTCLAS
PROC STORGRP
```

END

FILTLIST Statement

The FILTLIST statement is a definition list that you can use when testing variables in an ACS routine. You define the information that you want to include and exclude in the list using the INCLUDE and EXCLUDE keywords. Then you can compare read-only variables to items in the list using IF-THEN and SELECT-WHEN statements, without having to write elaborate AND and OR combinations.

FILTLIST is a definition statement that simplifies comparison operations. It is not an execution statement, and it does not change the value of any variables.

Because a FILTLIST can contain only literal values, you can only compare it to literal read-only variables. This excludes the numerically valued &NQUAL, &NVOL, &SIZE, &MAXSIZE, &MEMNQUAL, and &RETPD read-only variables from FILTLIST comparisons.

You must define a FILTLIST before you reference it in the body of an ACS routine.

FILTLIST *name* [<INCLUDE(*list*)>] [<EXCLUDE(*list*)>]

- *name* is mandatory and can be up to 31 alphanumeric characters in length. You can also use an underscore, _, but it cannot be the first character. In the FILTLIST, you can optionally precede the name with an ampersand, & When referring to the FILTLIST in the body of the routine, you must always precede the FILTLIST name with an ampersand.
- You must specify INCLUDE, EXCLUDE, or both in the FILTLIST statement. If a list item satisfies both the INCLUDE and EXCLUDE criteria, EXCLUDE takes precedence and prevents the item from being included in the list.
- *list* can contain literals, simple masks, and data set masks. You can specify up to 255 entries in the INCLUDE or EXCLUDE lists.

Figure 117 shows an example of coding a FILTLIST statement:

```
PROC STORCLAS

  FILTLIST VLIST2 INCLUDE(DBX*, TSO*) EXCLUDE('DBX191', 'TSO256')

  IF &ALLVOL = &VLIST2 THEN
      (some action)

END
```

Figure 117. Example of a FILTLIST Statement

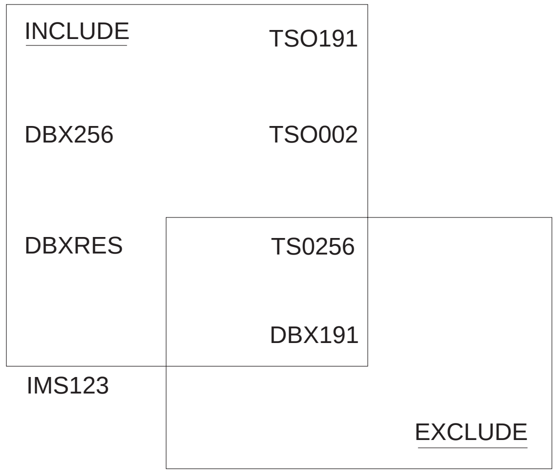

Figure 118. Using the INCLUDE and EXCLUDE

In the environment shown in Figure 118, the value of the IF statement is *true* for any of the following volume serials:

TSO191
TSO002
DBX256
DBXRES

The value of the IF statement is *false* for the following volume serials, because they match the EXCLUDE filter criteria:

TSO256
DBX191

SET Statement

The SET statement assigns values to the read-write variables. The values can be the names of constructs or RACF derived defaults (&DEF_DATACLAS, &DEF_STORCLAS, or &DEF_MGMTCLAS). You can assign one name to &DATACLAS, one name to &STORCLAS, and one name to &MGMTCLAS, but you can assign a list of up to 15 names to &STORGRP

The names can be from one to eight characters long, and they must be enclosed in single quotation marks. The individual names belonging to a &STORGRP must be enclosed in single quotation marks and separated by commas. A name must begin with either an alphabetic or national ($, @, #) character, and the remaining characters can be alphabetic, numeric, or national.

In the body of an ACS routine, you can set only the value of the read-write variable identified in the PROC statement. You cannot set a read-write variable equal to another read-write variable, a FILTLIST variable, or a read-only variable (except for the RACF-derived values). For example, the following is not valid:

```
SET &STORCLAS = &PGM
```

You can assign a null value to any of the read-write variables except for &STORGRP. You can assign a null value by specifying two single quotation marks with nothing between them ('').

```
SET read-write variable = value
```

where:

- *read-write variable* is a mandatory value that can be &DATACLAS, &STORCLAS, &MGMTCLAS, or &STORGRP.
- You can specify EQ in place of the equals sign, =.
- *value* can be one name, a null value, a RACF read-only variable name, or a list of names for the storage group ACS routine. Table 21 summarizes the possible assignments of *value*.

Table 21. Read-Write Variable Assignments

	One Name in Single Quotation Marks	Null Value	RACF Read-Only Variable Name	List of Quoted Names Separated by Commas
Storage group	x			x
Management class	x	x	x	
Storage class	x	x	x	
Data class	x	x	x	

```
SET &STORCLAS EQ 'SCNORM'
SET &STORCLAS = 'SCNORM'
SET &DATACLAS = &DEF_DATACLAS          /* RACF read-only variable */
SET &MGMTCLAS = 'SCRATCH5'
SET &STORGRP = 'SG1','SG2','SG3'      /* list of values          */
SET &DATACLAS = ''                    /* null value assignment   */
```

You must have RACF installed and ACSDEFAULT(YES) specified in the IGDSMSxx to assign the &DEF_DATACLAS, &DEF_STORCLAS, or &DEF_MGMTCLAS. Otherwise a null value is assigned to a read-write variable.

DO Statement

You can group a collection of ACS language statements using a DO statement paired with an END statement. The DO statement can follow an IF-THEN clause, an ELSE clause, or a SELECT-WHEN group.

DO *<group of statements>* **END**

group of statements can consist of zero or more ACS language statements.

Figure 119 shows an example of a DO statement:

```
IF &HLQ='PAYROLL' THEN
 DO
  WRITE 'No Payroll allowed'
  EXIT CODE(1)
 END
```

Figure 119. Example of a DO Statement

IF Statement

Use the IF statement for conditional statement execution. You must always follow an IF statement with a THEN clause. The THEN clause can be a single statement or a DO-END group of statements. You can optionally follow a THEN clause with an

ELSE clause. If you specify the ELSE clause, you must follow it with either a single statement or a DO-END group of statements. If you want to specify the ELSE clause but do not want to follow it with an executable statement, follow it with an empty DO-END pair.

If the result of the IF statement comparison is *true*, the THEN clause is executed. If the result is false, the ELSE clause is executed. If you omit the ELSE, processing continues with the next sequential statement after the THEN clause.

IF *relational expression* THEN *clause* <ELSE *clause*>

- *relational expression* can be a single comparison or it can be multiple comparisons joined by Boolean operators.
- THEN is a mandatory keyword.
- *clause* can contain a single statement, a DO-END group of statements, or a SELECT statement.
- ELSE is an optional keyword.
- An ELSE *clause* is mandatory if you specify the ELSE keyword.

Note: For information on what constitutes a valid relational expression, see “Comparison Rules” on page 273.

Figure 120 shows two examples of IF statements:

```
Example of a specified ELSE:          Example of a null ELSE:

IF &DSOWNER = 'BILL' THEN             IF &DSOWNER = 'BILL' THEN
  SET &STORCLAS = 'ONE'                     SET &STORCLAS = 'ONE'
ELSE                                  ELSE
  SET &STORCLAS = 'TWO'                     DO END
                                      SET &STORCLAS = 'TWO'
```

Figure 120. Examples of IF Statements

The statement on the left sets &STORCLAS equal to ONE or TWO, depending on the value of &DSOWNER

The statement on the right sets &STORCLAS equal to TWO, regardless of the value of &DSOWNER If &DSOWNER equals 'BILL', then &STORCLAS is set to 'ONE'. The ELSE clause is skipped, and execution falls to the next statement, which changes &STORCLAS to 'TWO'. If &DSOWNER does not equal 'BILL', execution falls to the ELSE, which results in no assignment to &STORCLAS Then execution proceeds to the next sequential statement, which sets &STORCLAS to 'TWO'.

SELECT Statement

Use the SELECT statement to write conditional statements in sequential form rather than IF-THEN-ELSE form. A SELECT statement consists of a SELECT keyword, one or more WHEN clauses, an optional OTHERWISE clause, and an END statement. You can specify the SELECT statement in one of two forms. In the first form shown below, you include the variable being tested after the SELECT statement. In the second form shown below, you include the variable being tested after the WHEN keyword.

The first true WHEN condition is executed, and the remaining WHEN conditions are ignored. If none of the WHEN conditions is true and there is an OTHERWISE clause, then the OTHERWISE action is taken.

```
SELECT (variable)
  WHEN (value) <action>
                .
                .
                .

 <WHEN (value) <action>>
 <WHEN (value) <action>>
  <OTHERWISE <action>>
END
```

where:

- *variable* can be any read-only or read-write variable.
- At least one WHEN keyword is mandatory.
- *value* can be a constant or a FILTLIST name.
- *action* can be an ACS statement, SELECT group, or a DO END group.
- OTHERWISE is an optional keyword.

```
SELECT (&DSOWNER)
  WHEN ('IBMUSER1') SET &STORCLAS = 'PAYROLL'
  WHEN ('IBMUSER2') SET &STORCLAS = 'TEST'
  WHEN ('IBMUSER3') SET &STORCLAS = 'DEVELOP'
  OTHERWISE      SET &STORCLAS = 'NORMAL'
END
```

SELECT WHEN (*relational expression*) <*action*> . . .

<WHEN (*relational expression*) <*action*>>
<WHEN (*relational expression*) <*action*>>

<OTHERWISE <*action*>> END

- At least one WHEN keyword is mandatory.
- *relational expression* can be a single comparison or it can be multiple comparisons joined by Boolean operators.
- *action* can be an ACS statement, SELECT group, or a DO END group.
- OTHERWISE is an optional keyword.

Figure 121 shows an example of coding a SELECT statement:

```
SELECT
  WHEN (&DSOWNER = 'IBMUSER') SET &STORCLAS = 'PAYROLL'
  WHEN (&DSOWNER = 'IBMUSER2')

     IF &ACCT_JOB = '1234' THEN
       SET &STORCLAS = 'TEST'
     ELSE
       SET &STORCLAS = 'EVERYONE'

  WHEN (&DSTYPE = 'TEMP')  SET &STORCLAS = '
  WHEN (&HLQ    = 'CADAM')   SET &STORCLAS = '
  OTHERWISE                  SET &STORCLAS = 'COMMON'
END
```

Figure 121. Example of a SELECT Statement

EXIT Statement

The EXIT statement immediately terminates the operation of an ACS routine.

EXIT <CODE(*n*)>

- CODE is an optional keyword.
- *n* is an exit code. A nonzero value for *n* causes the subsequent ACS routines to be skipped and the allocation to fail with no explicit value assigned to the read-write variable in the ACS routine. If you do not specify a value for *n*, it assumes the default value of zero.

Figure 122 shows an example of an EXIT statement:

```
PROC STORCLAS
  FILTLIST SECVOL INCLUDE(PAY*, REC*) EXCLUDE('PAYR20', 'REC195')
  FILTLIST VALID_UNITS  INCLUDE('3380','3390','SYSDA','')
  IF &UNIT ¬= &VALID_UNITS THEN

     DO
        SET &STORCLAS = '
        EXIT
     END

  IF &ALLVOL ¬= &SECVOL THEN EXIT CODE(22)

END
```

Figure 122. Example of an EXIT Statement

Note: The example above shows that pre-3380 devices are not supported on DFSMS/MVS 1.1 systems. However these earlier devices are still supported on MVS/DFP 3.1.1 and prior systems. The 3330V (MSS) is no longer supported on any system.

If the first IF statement is *true* (&UNIT does not match any unit named in the VALID_UNITS filter criteria), then execution of this ACS routine terminates immediately. Allocation proceeds, because the exit code is zero, the default.

If the second IF statement is *true* (none of the input volumes match the SECVOL FILTLIST criteria), then execution of this ACS routine terminates immediately and the allocation fails. The value for CODE, 22, is set and displayed as part of the allocation failed error message written to the end user.

WRITE Statement

Use the WRITE statement to issue a message to an end user at execution and allocation time. With the WRITE statement, you might notify end users that you are removing a particular storage class, you might inform end users that they lack sufficient authority to use a particular management class, or you might tell an end user that you have moved a tape data set to DASD.

In a TSO/E ALLOCATION environment, the text of the WRITE message displays only if the allocation fails.

With DFSMSdss, the text of the WRITE message is never displayed.

Under certain conditions related to data set stacking, SMS invokes ACS routines more than once. Consequently, you might want to take special care when using WRITE statements in order to avoid duplicates in the job log.

A WRITE message can contain up to 110 characters of text and variables. The message substitutes the value of a variable for a variable name. With the exception of &ANYVOL and &ALLVOL, you can use any of the read-only variables in write statements. All numerical values are in hexadecimal when displayed in write statements.

You must enclose the message in single quotation marks. If you want a single quotation mark to be part of the message, use two single quotation marks to represent it.

This: **Appears as this:**

WRITE 'This line''s short.'
This line's short.

You can use continuation characters (+, -) to continue text onto a subsequent line. The closing single quotation mark signifies the end of text.

A nine-character system message id and a single blank character precede your message to the end user. At execution and allocation time, an end user can receive a maximum of five messages. If any more messages are generated, a sixth and final message indicates that additional messages have been generated, but the additional messages are not displayed.

WRITE '*message*'

message is written with the end user's job messages.

Assuming the value of &STORCLAS is 'SC1:', the following WRITE message

```
WRITE 'WARNING - &STORCLAS SPECIFIED (' &STORCLAS ') IS NOT ALLOWED'
```

displays as:

```
IGD01005I WARNING - &STORCLAS SPECIFIED (SC1) IS NOT ALLOWED
```

END Statement

The END statement concludes an ACS routine, a DO group, or a SELECT statement. Figure 123 shows an example of an END statement:

```
PROC STORCLAS
 (source code)
END
```

Figure 123. Example of an END Statement

Sample ACS Routine

The following example illustrates some techniques for using the ACS routines.

```
PROC STORCLAS

/*******************************************************************************************/
/* THIS IS THE PRODUCTION SELECTION SPECIFICATION FOR SETTING STORCLAS                    */
/*******************************************************************************************/

FILTLIST DBVOLS INCLUDE(IMS*,DB2*)                                /* ALL DATABASE VOLUMES */
                EXCLUDE('IMS053','DB2007')

FILTLIST DBJOBS INCLUDE(IMS*,PROD*,ACCT*)                           /* ALL DATA BASE JOBS */

FILTLIST VALID_UNITS
INCLUDE('3330','3340','3350','3375','3380','3390','SYSDA','')      /* VALID UNITS FOR SMS */

IF &UNIT ^= &VALID_UNITS

   THEN DO
           SET &STORCLAS = '
           WRITE 'INVALID UNIT TYPE FOR SMS ALLOCATION'
           EXIT
        END

SELECT

       WHEN (&DSN = SYS1.**)                                               /* SYSTEM DATA */
            SET &STORCLAS = 'SYSTEM'

       WHEN ((&ALLVOL = &DBVOLS) && (&JOB = &DBJOBS))                    /* DATABASE DATA */
            SET &STORCLAS = 'DBPOOL'

       WHEN ((&DSN(3) = 'CLEAR') | (&ANYVOL ^= TSO*))                     /* NON-SMS DATA */
            SET &STORCLAS = '

       WHEN (&DEF_STORCLAS ^= '')                                   /* IF DEFAULTS EXIST */
            SET &STORCLAS = &DEF_STORCLAS;

       OTHERWISE SET &STORCLAS = 'COMMON'                                /* ALL OTHER DATA */

END                                                                 /* END STORCLAS PROC */
```

Figure 124. Production ACS Routine for Storage Class

The FILTLIST VALID_UNITS INCLUDE statement in Figure 124 does not contain the latest devices. Update the FILTLIST VALID_UNITS INCLUDE statement when new devices are installed at your installation.

Note: If the null unit (illustrated in the FILTLIST VALID_UNITS INCLUDE statement in Figure 124), is not in the valid DASD units FILTLIST, and a null storage class is assigned to allocations that do not have a valid unit (units in the DASD units FILTLIST), then you cannot manage VSAM data sets allocated using IDCAMS.

Appendix A. ISMF Command and Line Operator Reference Summary

The abbreviations in the Application column of these tables represent the following:

ACS Automatic class selection

AG Aggregate groups

CDS Control data set

DC Data class

DS Data set

DV Optical drive

DVOL DASD volume

LA List

LB Optical library

MC Management class

OVOL Mountable optical volume

SC Storage class

SG Storage group

TL Tape library

TVOL Mountable tape volume

Note: If you specify an equal sign after any DFSMSdss or DFSMShsm line operator, processing occurs in ’last-use mode’ which recalls the last values entered for that particular line operator or command and DFSMSdfp does not display an entry panel.

The following tables list the ISMF commands and line operators available to storage administrators. See *DFSMS/MVS Using ISMF* for the line operators and commands available to the end user.

Table 22. ISMF Commands

Command	Minimum Abbreviation	Description	Application	Source
ACTIVATE	AC	Copy the contents of an SCDS into an ACDS and activate it, or activate an existing ACDS.	CDS	DFSMSdfp
ALTER	AL	Alter the use attribute, storage group, shelf location or owner information (or any combination of these) for all tape volumes currently showing in the volume list.	OVOL, TVOL, TL	DFSMSdfp

Table 22. ISMF Commands (continued)

Command	Minimum Abbreviation	Description	Application	Source
AUDIT	AU	Verify the location of volumes in 3995 optical libraries or 3495 tape libraries.	OVOL, TVOL, TL	DFSMSdfp
BOTTOM	BOT	Scroll to the bottom of the entries.	All except ACS and CDS	DFSMSdfp
CANCEL	CA	Return to the previous dialog without performing any of the current dialog functions.	DC, SC, MC, SG, ACS, CDS, AG, LB, DV, LA	DFSMSdfp
CLEAR	CLE	Reset line operator history.	DS, DVOL, DC, SC, MC, SG, AG, LB, DV, OVOL, LA, TVOL, TL	DFSMSdfp
CLEAR ALL	CL ALL	Clear all pages on selection entry panels and filter panels.	DS, DVOL,	DFSMSdfp
CLEAR PAGE	CL PA	Clear the current page on selection entry panels and filter panels.	DS, DVOL, OVOL, LA, TVOL	DFSMSdfp
CLEAR PAGEx	CL PAGEx	Clear a page on selection entry panels and filter panels where x is the page number.	DS, DVOL,	DFSMSdfp
COMPRESS	COM	Reclaim embedded unused space from a list of PDSs.	DS	DFSMSdss
COPY	COP	Copy a list of data sets to a DASD volume of like or unlike device type.	DS	DFSMSdss
DOWN n	DO n	Scroll forward the number of list entries specified by n. DOWN MAX scrolls to the bottom of the entries.	All except ACS and CDS	DFSMSdfp
DSUTIL	DSUTIL	Invoke PDF data set utility functions.	DS	DFSMSdfp
DUMP	DU	Dump data sets to tape, DASD, or mass storage volumes.	DS	DFSMSdss
END	END	Exit the current ISMF function or panel and return to the previous panel.	All	DFSMSdfp
ERTB	ER	Display the ISMF Error Table.	All	DFSMSdfp

Table 22. ISMF Commands (continued)

Command	Minimum Abbreviation	Description	Application	Source
FILTER	FIL	Tailor the list to include only specific entries.	DS, DVOL, LA	DFSMSdfp
FILTER CLEAR	FIL C	Clear the filter entries but bypass the entry panel.	DS, DVOL, LA	DFSMSdfp
FIND	FIN	Find a specific data column.	All except ACS and CDS	DFSMSdfp
FOLD	FO	Extend the data set name data column.	DS	DFSMSdfp
HELP	HELP	Request the help panels associated with the current panel.	All	DFSMSdfp
LEFT	L	Scroll left the specified number of columns.	All except ACS and CDS	DFSMSdfp
LIBRARY	LIBRARY	Invoke PDF library utility.	DS	DFSMSdfp
LISTPRT	LISTP	Prints generated or saved ISMF lists	All that can generate a list	DFSMSdfp
PROFILE	P	Invoke the ISMF profile.	All except the ISMF Profile Application, the ISMF Menu panel or an abend panel	DFSMSdfp
QSAVE	QS	Save query criteria	DS, DVOL	NaviQuest
QRETRIEV	QR	Retrieve query criteria	DS, DVOL	NaviQuest
REFRESH	REF	Display the updated list.	All except ACS and CDS	DFSMSdfp
RELEASE	REL	Free unused space at the end of each of the data sets in a list.	DS	DFSMSdss
RESHOW	RESH	Redisplay all list entries removed by the HIDE line operator.	All except ACS and CDS	DFSMSdfp
RESTORE	REST	Restore data sets that have been dumped by DFSMSdss.	DS	DFSMSdss
RETURN	RETURN	Return to the primary ISMF option menu.	All	DFSMSdfp
RIGHT	RI	Scroll right the specified number of columns.	All except ACS and CDS	DFSMSdfp
SAVE	SA	Save a copy of the current list in the ISPF output table.	DS, DVOL, OVOL, DC, SC, MC, SG, LB, DV, AG, TVOL, TL	DFSMSdfp

Table 22. ISMF Commands (continued)

Command	Minimum Abbreviation	Description	Application	Source
SORT	SO	Organize lists based on entries in specific data columns.	All except ACS and CDS	DFSMSdfp
TOP	TOP	Scroll to the top of the entries.	All except ACS and CDS	DFSMSdfp
TSO Commands and CLISTs		Invoke TSO commands and CLISTs.	All	DFSMSdfp
UP	U	Scroll backward the specified number of entries. UP MAX scrolls to the top of the entries.	All except ACS and CDS	DFSMSdfp
VALIDATE	V	Check the completeness and consistency of an entire SCDS.	CDS	DFSMSdfp
VIEW	VI	Choose the display order of the columns on list panels.	All except ACS and CDS	DFSMSdfp

Table 23. ISMF Line Operators

Line Operator	Minimum Abbreviation	Description	Application	Source
ABACKUP	AB	Specify the parameters for an ABACKUP command and issue the command to DFSMShsm.	AG	DFSMShsm
ALTER	AL	Change the name of a management class, storage class, use attribute, storage group, shelf location, or owner information (or any combination of these) for all volumes associated with a data set or change the use attributes of an SMS class or storage group.	DS, DC, SC, MC, SG, AG, LB, DV, TVOL, OVOL, TL	DFSMSdfp
ANALYZE	ANAL	Examine the device or data on a volume to determine if any errors exist.	DVOL	ICKDSF
AUDIT	AU	Verify the location of optical volumes in IBM 3995 optical and 3495 tape libraries.	OVOL, LB, TVOL, TL	DFSMSdfp

Table 23. ISMF Line Operators (continued)

Line Operator	Minimum Abbreviation	Description	Application	Source
BROWSE	B	View a sequential data set or a member of a PDS.	DS	DFSMSdfp
BUILDIX	BUI	Change a volume from MVS format VTOC(OSVTOC) to an indexed format VTOC(IXVTOC) or vice versa.	DVOL	ICKDSF
CATLIST	CATLIST	Invoke IDAMS LISTCAT and browse the output.	DS	DFSMSdfp
CLIST	CLI	Call a TSO CLIST.	DS, DVOL, OVOL, TVOL, TL	DFSMSdfp
COMPRESS	COM	Reclaim embedded unused space from a PDS.	DS, DVOL	DFSMSdss
CONDENSE	CON	Free unused space at the end of a data set; compress a PDS.	DS	DFSMShsm
CONTROL	CONT	Reset a device that has been WRITE INHIBITed, reset an indefinite status condition, or clear a fence status of a path or a device or both.	DVOL	ICKDSF
CONVERTV	CONV	Convert DASD volume(s) into SMS or out of SMS.	DVOL	DFSMSdss
COPY	COP	Copy one SMS class or storage group to another SMS class or storage group in the same or in a different SCDS; or copy a data set volume to a DASD volume.	DS, DVOL, DC, SC, MC, SG, AG, LB, DV, TL	DFSMSdfp or DFSMSdss
DEFRAG	DEFR	Reduce free-space fragmentation on a DASD. volume	DVOL	DFSMSdss
DELETE	DEL	Delete an SMS class or storage group, or delete an online, backup, or DFSMShsm-migrated data set.	DS, DC, SC, MC, SG, AG, LB, DV, LA, TL	DFSMSdfp or DFSMShsm
DISPLAY	DI	Display an SMS class and its attributes.	DC, SC, MC, AG, LB, DV, TL	DFSMSdfp

Table 23. ISMF Line Operators (continued)

Line Operator	Minimum Abbreviation	Description	Application	Source
DUMP	DU	Dump a data set or volume to tape or DASD.	DS, DVOL	DFSMSdss
EDIT	E	Edit a sequential data set or member of a PDS.	DS	DFSMSdfp
EJECT	EJ	Eject an optical or tape volume from a library to an output station.	OVOL, TVOL, TL	DFSMSdfp
ERASE	ERA	Delete an SMS class or storage group, or delete an online, backup, or DFSMShsm-migrated data set.	DS, DC, SC, MC, SG, AG, LB, DV, LA	DFSMSdfp or DFSMShsm
HALTERDS	HA	Change the number of backup versions of a data set; change frequency of backup.	DS	DFSMShsm
HBACKDS	HBA	Create a backup version of a data set.	DS	DFSMShsm
HBDELETE	HBD	Delete backup versions of a data set.	DS	DFSMShsm
HDELETE	HDE	Delete a migrated data set.	DS	DFSMShsm
HIDE	HI	Remove a list entry from display.	All	DFSMSdfp
HMIGRATE	HM	Migrate a data set to DFSMShsm level one or level two volume.	DS	DFSMShsm
HRECALL	HRECA	Recall a data set that has been migrated by DFSMShsm.	DS	DFSMShsm
HRECOVER	HRECO	Recover a backup version of a data set.	DS	DFSMShsm
INIT	INI	Initialize a volume.	DVOL	ICKDSF
INSPECT	INS	Detect defects in volume track surface.	DVOL	ICKDSF
INSTALL	INST	Install an HDA replacement and physical movement of IBM DASD.	DVOL	ICKDSF
LIST	LI	Retrieve a list that was saved with the SAVE command.	LA	DFSMSdfp
LISTSYS	LISTS	List the systems associated with a given storage group.	SG	DFSMSdfp

Table 23. ISMF Line Operators (continued)

Line Operator	Minimum Abbreviation	Description	Application	Source
LISTVOL	LISTV	List the volumes associated with a given storage group, optical library or tape library.	SG, LB, TL	DFSMSdfp
MESSAGE	MES	Display message text for the last operation performed on a list entry.	All	DFSMSdfp
RAUTH	RAUTH	Provide remote access codes.	DVOL	DFSMSdfp
RECOVER	REC	Recover a backup copy of all the objects on a mountable optical volume.	OVOL	DFSMSdfp
REFORMAT	REF	Change the volume serial number or owner ID of a volume.	DVOL	ICKDSF
RELEASE	REL	Free unused space at the end of data sets.	DS, DVOL	DFSMSdss
REMAP	REM	Reconstruct inventory in a real IBM 3995 optical library.	LB	DFSMSdfp
REPEAT	=	Repeat the last operator command that was implemented.	All	DFSMSdss DFSMSdfp
RESTORE	REST	Restore data sets that have been dumped by DFSMSdss.	DS, DVOL	DFSMSdss
REVAL	REV	Perform track validation of medial initialization.	DVOL	ICKDSF
SECURITY	SE	Invoke a RACF panel to protect data sets, storage classes, or management classes.	DS, SC, MC, LB, DV, TL	RACF
SETCACHE	SETC	Manage storage control characteristics.	DVOL	DFSMSdfp
SORTREC	SOR	Invoke DFSORT to sort data set records.	DS	DFSORT
STATUS	ST	Allows the display of up to 32 SMS and MVS volume statuses.	DVOL	DFSMSdfp
TSO Commands and CLISTs		Invoke TSO commands and CLISTs.	All	DFSMSdfp
VTOCLIST	VTOCLIST	Invoke IEHLIST LISTVTOC.	DS	DFSMSdfp

Appendix B. SETCACHE Functions and Device Information

This appendix shows you:

- Tasks that maintain the media characteristics
- Ways to modify DASD storage control
- Ways to provide access to IBM Service Representatives at remote locations.

To perform these tasks, you enter line operators or list commands on data set or volume lists and then complete data entry panels. Refer to your online help panels or *DFSMS/MVS Using ISMF*.

Tasks That Maintain the Media Characteristics of a Volume

You can use ISMF to examine and manage the characteristics of your data stored on DASD volumes. Eight line operators are available to perform the following media maintenance tasks:

- Inspect a subset of a volume for magnetic surface defects. ISMF also provides options to help you remedy the problem.
- Examine a drive and your data to determine if errors exist. This function helps you distinguish between errors caused by drive problems and errors caused by media problems.
- Initialize a DASD volume for use in an MVS system.
- Change the volume label of a DASD volume. You can specify a new volume serial number or a new owner ID.
- Generate a job stream that is used to change a volume from an MVS format VTOC(OSVTOC) to an indexed format VTOC(IXVTOC) or vice versa.
- Reset a device that has been WRITE INHIBITed, reset an indefinite status condition, or clear a fence status of a path, a device, or both.
- Generate a job stream that is used to perform the procedures necessary for installation, head-disk assembly (HDA) replacement, and physical movement of IBM DASD.
- Perform the track validation functions of medial initialization with the problem determination and data verification functions of the ANALYZE command, and also the INSPECT functions if required.

For information on the ICKDSF functions that ISMF uses to perform these functions see *ICKDSF User's Guide and Reference*.

Ways to Modify DASD Storage Control Characteristics

You can use the SETCACHE line operator to control the caching in a storage control unit.

Table 24. Summary of the SETCACHE Line Operator

Task	Function	Scope	Action
Manage storage control unit characteristics	SETCACHE	Volume and storage control unit	Modifies the caching status, duplexing status, or causes data to be destaged in the storage control units associated with specific DASD volumes. ISMF performs these functions in the foreground using a TSO command or in the background using the ISMF job submission facility.

You can use the SETCACHE line operator from the Volume List panel to modify the storage control unit characteristics. The following characteristics can be modified with the SETCACHE line operator:

- Performance characteristics, which control the caching of read or write requests. You can modify these characteristics to reduce the frequency of access to DASD.
- Availability characteristics, which control the automatic duplication of the device activity onto a secondary volume. You can also modify these characteristics to move the contents of one volume to another while the data is active.
- Resources characteristics, which control how data that is stored in the cache is destaged or discarded. You can directly manage data in cache and nonvolatile storage.

Enter the SETCACHE line operator against a particular volume from the Volume List panel. You can invoke SETCACHE with a PERFORMANCE, AVAILABILITY, or RESOURCES parameter depending on the characteristic you would like to modify. You can also enter SETCACHE without a parameter. On a 3990 Storage Control unit with cache, ISMF takes you to the SETCACHE Features Entry panel. From this panel you can select the characteristic you would like to modify. ISMF takes you to either the PERFORMANCE, AVAILABILITY, or RESOURCES Entry panel. On the Model 3880-13/23, ISMF takes you to the PERFORMANCE Entry panel where you can do your modifications.

The specific features that you can modify depend on the model of the storage control unit that you are working with. You must also have storage administration authorization to use the SETCACHE line operator.

In order to use the SETCACHE function under ISMF, you must modify the IKJTSOxx member in SYS1.PARMLIB. IDCAMS must be added to AUTHPGM NAMES, and SETCACHE must be added to AUTHCMD NAMES.

Modifying Caching Characteristics

Use the Setcache Performance Entry panel to modify the caching characteristics of the storage control units. If you are using a Model 3990 Model 3 Storage Control, you have support for the caching of both read and write requests. If you are using a 3880-13 or 3880-23 Storage Control, you have support for the caching of read requests. In the terminology of the SETCACHE line operator, the caching of read requests is controlled by the READ CACHE or READ and SYSTEM CACHE functions. The caching of write requests is controlled by the DASD FAST WRITE, NON-VOLATILE STORAGE (NVS), and CACHE FAST WRITE functions.

Read Caching/Read and System Caching

You can establish or terminate the READ CACHE or READ and SYSTEM CACHE functions to control the caching of read requests. Read caching is performed on a track basis. When you need data from a volume, a read request is made to the storage control unit The storage control unit reads the data from the volume and sends it back to you. At the same time, a copy of all the data on the track that the storage control unit accesses is saved in cache that resides in the storage control unit. Any subsequent requests that you make for data on the track can be satisfied from the cache. Read performance is improved when the data for your read requests come from the cache.

You can specify that the storage control unit perform this type of read caching on a volume level. Caching can be turned on and off for individual volumes. This feature gives you greater flexibility to manage the volumes that are using the caching resources.

Write Caching

If your storage control unit supports the caching of write requests, you can use the SETCACHE line operator to establish or terminate the DASD FAST WRITE and CACHE FAST WRITE functions. Note that read and system caching must be active in order for fast write (both DASD and CACHE) to be in effect.

DASD FAST WRITE provides support for the caching of write requests. NON-VOLATILE STORAGE (NVS) provides backup storage for this function. When you write data out to a volume, a write request is made to the storage control unit. The storage control unit accepts the data and puts it in the storage control unit cache. Later, the storage control unit schedules the data for writing.

When the data is stored in the storage control unit cache, it is also written to NVS. NVS is a buffer backed up by battery power that can maintain data up to 48 hours. NVS provides insurance against the loss of data written to the storage control unit cache and not yet sent to DASD.

The CACHE FAST WRITE feature allows part of the subsystem storage to be set aside as work space. Data written out in CACHE FAST WRITE mode is not staged for writing to DASD unless you make an explicit request or the system needs to re-use some of the cache space. CACHE FAST WRITE is for temporary work files only because data in cache is eventually discarded.

Modifying Duplexing Characteristics

Use the Setcache Availability Entry panel to manage the availability of a duplex pair of devices that control the duplication and recovery of data. The AVAILABILITY characteristics affect only data that is on DASD.

To establish a duplex pair, you must specify:

- A primary and a secondary device
- The type of synchronization between the two devices

You can specify the rate of copy in the synchronization. Once duplexing is established, the storage control unit writes data to the primary device and simultaneously places a copy of it in cache along with control information in NVS. At a later time, the data in cache is written out to the secondary device, and the control information in NVS is updated.

If duplexing is interrupted, the dual copy pair is suspended. The storage control still writes information to NVS, but data is written to only the primary device. When duplexing is resumed, the secondary device is placed back in synchronization by the storage subsystem.

You also can use the Setcache Availability Entry panel to move the contents of one device to another. The panel allows you to establish a duplex-pair between two devices to copy data. The pair is immediately broken when all data has been copied. The original device is then available for maintenance.

Modifying Destaging Characteristics

Use the Setcache Resources Entry panel to directly manage the data that has already been written to the cache and NVS. This function of the SETCACHE line operator allows you to:

- Destage data
- Discard pinned data
- Set all the volumes and the subsystem back to their default status

Once data has been written to the cache or NVS, the Resources Entry panel allows you to write the data to DASD. If this destaging process fails, the data is pinned in cache or NVS. The Resources Entry panel allows you to discard this pinned data. All the volumes attached to a storage subsystem along with the subsystem itself can be set back to their default status by re-initializing the storage subsystem. Note that re-initializing a subsystem would cause all duplex pairs to be lost and any fast write data in cache and NVS to be discarded.

Submitting Jobs

When you specify the SETCACHE line operator, ISMF takes you through the entry panels you need to complete given the functions you choose to perform. ISMF eventually asks you to specify whether the job is performed in the foreground or the background. In the foreground, ISMF uses the IDCAMS TSO SETCACHE support. ISMF subsequently re-displays updated versions of the entry panels with their new values after the job has executed. The list panels are not updated unless a REFRESH is done. For some functions, execution in foreground can occupy your TSO terminal for 20 minutes or more. In these cases, execution in the background is recommended.

When you specify that the SETCACHE job be performed in background, ISMF displays the IDCAMS Job Submission Entry panel. From this panel, you can submit the job or specify a data set where ISMF saves the job. The Job Submission Entry panel also optionally allows you to change the JCL that ISMF uses to run the job. At each level of the SETCACHE panels, the ISMF online help provides detailed descriptions for each of the fields.

Once ISMF takes you into the Setcache Entry panel, the online help provides detail descriptions for each of the fields.

Ways to Provide Remote Authorization Codes

The RAUTH line operator provides an easy way to acquire the remote access authorization codes for your IBM 3990 Storage Control units.

Table 25. Summary of the RAUTH Line Operator

Task	Function	Scope	Action
Provide remote access codes	RAUTH	Volume	Invokes a RAUTH display panel that shows the remote access authorization codes.

When you enter the RAUTH line operator from the Volume List panel, ISMF returns passwords on the RAUTH display panel.

Each password is valid for one hour. Once the password is displayed, ISMF does not display the password again. If you specify RAUTH again, ISMF returns new passwords.

Using these passwords, an off-site IBM Service Representative can log on to all models of the IBM 3990 Storage Control unit family (Models 1, 2, 3, and 6). The Service Representative can then provide help diagnosing and correcting problems.

You must have storage administrator authorization to use the RAUTH line operator. Because the standard support facilities of ISPF are available, for example, printing the screen, it is your responsibility to maintain the security of the passwords. Also, RACF DASDVOL alter authority is required over the volume the request is made against. ISMF online help provides a description of each of the fields on the RAUTH display panel.

Appendix C. Collecting Space and Capacity Planning Information

You can collect information on active and inactive data set space utilization and capacity planning by using the ISMF Data Collection application to produce a DCOLLECT job. DCOLLECT provides measurement data in a sequential data set (also called a “flat” file), which can be used as input to other applications such as billing and report formatting.

You can use it to produce measurement data on: active data sets, which are data sets that have not been backed up or migrated; inactive data sets, which are data sets that *have* been backed up or migrated; capacity planning, which concerns volume capacity and usage; and volumes. Information can be obtained on the following:

- Active data sets
- Volumes
- Migrated data sets (DFSMShsm)
- Backup data sets (DFSMShsm)
- DASD capacity planning (DFSMShsm)
- Tape capacity planning (DFSMShsm)

DCOLLECT is an access method services function.

Using the Data Collection Application

You can use ISMF panels to generate the job control language for the access method services DCOLLECT command. The job is expected to be executed while the DFSMS environment is active. To use these panels, select the Data Collection Application option on the ISMF Primary Option Menu for Storage Administrators.

Figure 125 on page 302 shows the Data Collection Entry panel, page 1. The fields on the entry panel for the Data Collection application are primed to collect data only for active data sets and volumes. You can also collect information on migration data, backup data and capacity planning data. You can specify Y, yes, in any combination of the fields under SELECT DATA COLLECTION OPTIONS.


```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
DGTDAD01                 DATA COLLECTION ENTRY PANEL             Page 1 of 3
Command ===>

Select Data Collection options:
  Data Set Information  . . . Y          (Y or N; Y requires volume(s) or
  Volume Information  . . . . Y          storage group(s) on next page)
  Migration Data  . . . . . . N          (Y or N)
  Backup Data . . . . . . . . N          (Y or N)
  Capacity Planning Data  . . N          (Y or N)
  SMS Data  . . . . . . . . . N          (Y or N)
Specify Output Data Set:                 (1 to 44 Characters)
  Data Set Name . . . . . . .
  Optional Password . . . . .            (Ignored if SMS-managed data set)
  Replace Contents  . . . . . N          (Y or N)
  Number of Data Sets . . . . 1          (1 to 99999999; new data set only)
Specify Input Data Set:                  (1 to 44 Characters)
  Migration Data Set Name . .
  Backup Data Set Name  . . .
  CDS Name  . . . . . . . . . 'ACTIVE'

Use ENTER to Perform Selection; Use DOWN Command to View next Entry Panel;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 125. Entry Panel for the Data Collection Application, Page 1 of 2

The following list explains the fields on page 1 of the Data Collection Entry panel.

DATA SET INFORMATION
If you want information collected on active data sets selected by storage group or volume, specify Y, yes, in this field. Otherwise, specify N, no. The default is Y.

VOLUME INFORMATION
If you want to collect information on active volumes selected by storage group or volume, specify Y, yes, in this field. Otherwise, specify N, no. The default is Y.

MIGRATION DATA
If you want to collect information on migration data, specify Y in this field. Otherwise specify N, no. If you specify Y, you must specify a migration data set name in the MIGRATION DATA SET NAME field. The default is N.

BACKUP DATA
If you want to collect information on backup data, specify Y in this field. Otherwise specify N, no. If you specify Y, you must specify a backup data set name in the BACKUP DATA SET NAME field. The default is N.

CAPACITY PLANNING DATA
If you want to collect information on capacity planning, specify Y in this field. Otherwise specify N, no. If you specify Y, you must specify both a migration data set name and a backup data set name. The default is N.

SMS DATA
Specify Y, yes, in this field if you want to collect SMS information. The default is N.

DATA SET NAME
You must enter the name of an output data set in this field. DCOLLECT puts the information it collects in the data set you specify. The data set can either be an existing data set or a data set that is allocated during the data collection job while DFSMS is active.

OPTIONAL PASSWORD
If the data set is not system-managed and it has a password, you must specify the password in the OPTIONAL PASSWORD field. Password protection is less secure than RACF protection. The data set must be cataloged. This field is ignored for system-managed data sets.

REPLACE CONTENTS
If you want DCOLLECT to overlay any data which is already contained in the output data set, specify Y, yes, in the REPLACE CONTENTS field. If you want DCOLLECT to append newly collected data to the data already contained in the output data set, specify N, no, in the REPLACE CONTENTS field. N, no, is the default, that is, data already contained in the output data set is not erased when the job is submitted. If the data set has never been written in, it does not matter whether you specify Y or N.

NUMBER OF DATA SETS
If you are specifying an output data set that is to be allocated during the data collection job, you should estimate the total number of data sets which reside on the volumes about which you are collecting data. Specify this number in the NUMBER OF DATA SETS field. This number is used to estimate the size of the output data set. If you save the JCL generated, instead of submitting it directly from this application, you can edit the DD statement for the output data set to change the space calculation.

MIGRATION DATA SET NAME
If you are collecting migration data or information for capacity planning, you must specify a data set name in the MIGRATION DATA SET NAME field. This data set must contain migration information. For DFSMShsm, this data set is the Migration Control Data Set (MCDS) for either the system you are running on or the system about which you want information. This data set must be accessible to the system running the DCOLLECT job.

BACKUP DATA SET NAME
If you are collecting backup data or information for capacity planning, you must specify a data set name in the BACKUP DATA SET NAME field. This data set must contain backup information. For DFSMShsm, this data set is the Backup Control Data Set (BCDS) for either the system you are running on or the system about which you want information. This data set must be accessible to the system running the DCOLLECT job.

When you are collecting capacity planning information, you must specify migration and backup data sets that are on the same DFSMShsm system.

If you are collecting information on active data sets or volumes, use the DOWN command to reach the next page of the Data Collection Entry panel shown in Figure 126 on page 304.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                          DATA COLLECTION ENTRY PANEL              Page 2 of 3
Command ===>

Specify Pool Storage Groups (Full Name)
   ==>                                ==>
   ==>                                ==>
   ==>                                ==>
   ==>                                ==>
   ==>                                ==>
Specify Volumes (Full, Partial, ******, or *)
   ==>           ==>           ==>           ==>           ==>
   ==>           ==>           ==>           ==>           ==>
   ==>           ==>           ==>           ==>           ==>
   ==>           ==>           ==>           ==>           ==>
   ==>           ==>           ==>           ==>           ==>
   ==>           ==>           ==>           ==>           ==>
   ==>           ==>           ==>           ==>           ==>
   ==>           ==>           ==>           ==>           ==>
   ==>           ==>           ==>           ==>           ==>
   ==>           ==>           ==>           ==>           ==>
Use ENTER to Perform Selection; Use UP Command to View previous Entry Panel;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 126. Entry Panel for the Data Collection Application, Page 2 of 2

On this panel, you must specify one or more volumes or storage groups about which DCOLLECT is to collect information.

If you specified Y, yes, in the DATA SET INFORMATION or VOLUME INFORMATION fields on the first page of the DCOLLECT entry panel, you must specify at least one volume or storage group on this panel.

SPECIFY VOLUMES

In this field, you can specify the names of the volumes from which information is collected. The information collected can be about the volumes themselves or about the active data sets they contain. You can specify a full volume name (for example, SYSTSO), a partial volume name (for example, SYST*), an asterisk to indicate all online volumes, or six asterisks to indicate the system residence (SYSRES) volume.

DCOLLECT ignores repeated entries. If you specify a volume more than once, DCOLLECT collects information on that volume only once. This includes volumes belonging to storage groups that you have specified. If you specify a single asterisk, DCOLLECT ignores all other entries in the SPECIFY VOLUMES field. Although you can specify only 50 volumes on this panel, you can edit the job control language produced to specify up to 255 separate volumes.

SPECIFY POOL STORAGE GROUPS

In this field, you can specify the names of pool storage groups from which information is collected. Information from all of the volumes in a specified storage group is collected. Enter only pool storage group names. Only pool storage groups are appropriate for data set or volume information collection. The information collected can be on the volumes themselves or on the active data sets they contain. You must specify the full name of a storage group. You cannot specify a partial name or an asterisk for a storage group name.

Although you can specify only 10 storage groups on this panel, you can edit the job control language produced to specify up to 255 storage groups.

If you are collecting information only on active data sets, DCOLLECT collects information on the active data sets on the volumes and, in the storage groups you specify. It does not collect information on other data sets on those volumes and in those storage groups.

If you are collecting information only on active volumes, DCOLLECT collects information on the volumes and in the storage groups you specify. It ignores information about the data sets on those volumes.

If you want to specify volumes to exclude, use DOWN command to reach the next page of the Data Collection Entry panel shown in Figure 127.

```
  Panel  Utilities  Scroll  Help
------------------------------------------------------------------------------
                          DATA COLLECTION ENTRY PANEL             Page 3 of 3
Command ===>

Specify Exclude Volumes (Full or Partial)
   ==>           ==>           ==>           ==>           ==>
   ==>           ==>           ==>           ==>           ==>











Use ENTER to Perform Selection; Use UP Command to View previous Entry Panel;
Use HELP Command for Help; Use END Command to Exit.
```

Figure 127. Entry Panel for the Data Collection Application, Page 3 of 3

When you have finished entering values on the Data Collection Entry panel, press ENTER to display the DCOLLECT Job Submission Entry panel. On this panel, you can choose to submit the job you just specified, or save the job to a data set, which you can then edit. You can edit the job statement on this screen before submitting the job.

The JCL produced uses the access method services execute statement that you have already set up in the profile as a default. You can specify that you want to edit the execute statement on the Data Collection Job Submission Entry panel. When you press ENTER, you can edit the execute statement.

Abbreviations

The following acronyms and abbreviations are defined as they are used in the DFSMS/MVS library. If you do not find the abbreviation you are looking for, see *Dictionary of Computing*, New York: McGraw-Hill, 1994.

This list may include acronyms and abbreviations from:

- The *American National Standard Dictionary for Information Systems* , ANSI X3.172-1990, copyright 1990 by the American National Standards Institute (ANSI). Copies may be purchased from the American National Standards Institute, 1430 Broadway, New York, New York 10018.
- The *Information Technology Vocabulary* developed by Subcommittee 1, Joint Technical Committee 1, of the International Organization for Standardization and the International Electrotechnical Commission (ISO/IEC JTC1/SC1).

ABARS. Aggregate Backup and Recovery Support.

ACB. Application control block.

ACDS. Active control data set.

ACS. Automatic class selection.

AOR. Application owning region.

BCDS. Backup control data set.

BCS. Basic catalog structure.

CCSID. Coded character set identifier.

CDRA. Character data representation architecture.

CDS. Control data set.

CEC. Central electronic complex.

CF. Coupling facility.

CFRM. Coupling facility resource manager.

CICS. Customer information control system.

COMMDS. Communications data set.

CTC. Channel-to-channel.

CVAF. Common VTOC access facility.

DASD. Direct Access Storage Device.

DB2. DATABASE 2.

DBMS. Database management systems.

DFSORT. Data facility sort.

DSCB. Data set control block.

DCME. Dynamic cache management enhanced.

DTR. Device transfer rate.

FOR. File owning regions.

GDG. Generation data group.

GSR. Global shared resources.

GTF. Generalized trace facility.

HCD. Hardware configuration definition.

HFS. Hierarchical file system.

IART. Initial access response time.

ICF. Integrated catalog facility.

ICKDSF. Device Support Facilities.

IDRC. Improved data recording capability.

IML. Initial machine load.

ISMF. Interactive Storage Management Facility.

ISPF. Interactive System Productivity Facility. An interactive base for ISMF.

IXVTOC. Indexed format volume table of contents.

JCL. Job control language.

LCS. Library control system.

LSR. Local shared resources.

MCDS. Migration control data set.

MIPS. Million instructions per second.

MSR. Millisecond response time.

MSS. Mass storage subsystem.

MVS. Multiple virtual storage.

MVSCP. MVS configuration program.

NSR. Non-shared resources

NVR. Non-VSAM volume record.

NVS. Nonvolatile storage.

OAM. Object access method.

OSMC. OAM storage management component.

OSVTOC. MVS format volume table of contents.

OVTOC. Optical volume table of contents.

PDF. Program development facility.

PDS. Partitioned data set.

PDSE. Partitioned data set extended.

PO. Partitioned organization.

PS. Physical sequential.

RACF. Resource authorization control facility.

RAID. Redundant array of independent disks.

RMF. Resource measurement facility.

RLS. Record-level sharing.

SCDS. Source control data set.

SCSI. Small computer system interface.

SDR. Sustained data rate.

SHCDS. Sharing control data sets.

SMS. Storage management subsystem.

SRM. System resource manager.

SPUFI. SQL processing using file input.

TSO. Time sharing option.

TSO/E. Time sharing option extensions.

VIO. Virtual input/output.

VSAM. Virtual storage access method.

VTOC. Volume table of contents.

VVDS. VSAM volume data set.

WORM. Write-once read many.

XCF. Cross-system coupling facility.

Glossary

A

access method services. A multifunction service program that manages VSAM and non-VSAM data sets, as well as integrated catalog facility (ICF) and VSAM catalogs. Access method services provides the following functions:

- defines and allocates space for VSAM data sets, VSAM catalogs, and ICF catalogs
- converts indexed-sequential data sets to key-sequenced data sets
- modifies data set attributes in the catalog
- reorganizes data sets
- facilitates data portability among operating systems
- creates backup copies of data sets
- assists in making inaccessible data sets accessible
- lists the records of data sets and catalogs
- defines and builds alternate indexes
- converts CVOLS and VSAM catalogs to ICF catalogs

ACS installation exit. User-written code, run after an ACS routine, that provides capabilities beyond the scope of the ACS routine.

ACS interface routine. This calls an ACS routine from an ACS installation-exit routine.

activate. To load the contents of an SCDS into SMS address space storage and into an ACDS, or to load the contents of an existing ACDS into SMS address space storage. This establishes a new storage management policy for the SMS complex.

active configuration. A copy of an activated SCDS, which controls storage management policies for the SMS complex.

active control data set (ACDS). A VSAM linear data set that contains an SCDS that has been activated to control the storage management policy for the installation. When activating an SCDS, you determine which ACDS will hold the active configuration (if you have defined more than one ACDS). The ACDS is shared by each system that is using the same SMS configuration to manage storage. See also *source control data set* and *communications data set*.

active data. (1) Data that can be accessed without any special action by the user, such as data on primary storage or migrated data. Active data also can be stored on tape volumes. (2) For tape mount management, application data that is frequently referenced, small in size, and managed better on DASD than on tape. Contrast with *inactive data*.

aggregate backup. The process of copying an aggregate group and recovery instructions so that a collection of data sets can be recovered later as a group.

aggregate group. A collection of related data sets and control information that have been pooled to meet a defined backup or recovery strategy.

automated tape library. A device consisting of robotic components, cartridge storage areas, tape subsystems, and controlling hardware and software, together with the set of tape volumes that reside in the library and can be mounted on the library tape drives. See also *tape library*. Contrast with *manual tape library*.

automatic backup. (1) In DFSMShsm, the process of automatically copying data sets from primary storage volumes or migration volumes to backup volumes. (2) In OAM, the process of automatically copying objects from DASD, optical, or tape volumes contained in an object storage group, to backup volumes contained in an object backup storage group.

automatic class selection (ACS) routine. A procedural set of ACS language statements. Based on a set of input variables, the ACS language statements generate the name of a predefined SMS class, or a list of names of predefined storage groups, for a data set.

automatic dump. In DFSMShsm, the process of using DFSMSdss automatically to do a full-volume dump of all allocated space on a primary storage volume to designated tape dump volumes.

automatic primary space management insert. In DFSMShsm, the process of deleting expired data sets, deleting temporary data sets, releasing unused space, and migrating data sets from primary storage volumes automatically.

automatic secondary space management. In DFSMShsm, the process of automatically deleting expired migrated data sets, deleting expired records from the migration control data sets, and migrating eligible data sets from migration level 1 volumes to migration level 2 volumes.

availability. For a storage subsystem, the degree to which a data set or object can be accessed when requested by a user.

B

backup. The process of creating a copy of a data set or object to be used in case of accidental loss.

backup control data set (BCDS). In DFSMShsm, a VSAM key-sequenced data set that contains information about backup versions of data sets, backup volumes, dump volumes, and volumes under control of the backup and dump functions of DFSMShsm.

base configuration. The part of an SMS configuration that contains general storage management attributes, such as the default management class, default unit, and default device geometry. It also identifies the systems or system groups that an SMS configuration manages.

basic catalog structure (BCS). The name of the catalog structure in the integrated catalog facility environment. See also *integrated catalog facility catalog*.

C

cache fast write. A storage control capability in which the data is written directly to cache without using nonvolatile storage. Cache fast write is useful for temporary data or data that is readily recreated, such as the sort work files created by DFSORT. Contrast with *DASD fast write*.

cache set. A parameter on storage class and defined in the base configuration information that maps a logical name to a set of CF cache structure names.

capacity planning. The process of forecasting and calculating the appropriate amount of physical computing resources required to accommodate an expected workload.

Channel-to-channel (CTC). A method of connecting two computing devices.

Character Data Representation Architecture (CDRA) API. A set of identifiers, services, supporting resources, and conventions for consistent representation, processing, and interchange of character data.

class transition. An event that brings about change to an object's service-level criteria, causing OAM to invoke ACS routines to assign a new storage class or management class to the object.

cluster. A data component and an index component in a VSAM key-sequenced data set; or a data component alone in a VSAM entry-sequenced data set.

Coded Character Set Identifier (CCSID). A 16-bit number that identifies a specific encoding scheme identifier, character set identifiers, code page identifiers, and additional coding required information. The CCSID uniquely identifies the coded graphic character representation used.

collection. A group of objects that typically have similar performance, availability, backup, retention, and class transition characteristics. A collection is used to catalog a large number of objects which, if cataloged separately, could require an extremely large catalog. All collections must be managed by SMS.

communications data set (COMMDS). The primary means of communication among systems governed by a single SMS configuration. The COMMDS is a VSAM linear data set that contains the name of the ACDS and current utilization statistics for each system-managed volume, which helps balance space among systems running SMS. See also *active control data set* and *source control data set*.

compatibility mode. For DFSMS/MVS, it is the mode of running SMS in which no more than eight names—representing systems, system groups, or both—are supported in the SMS configuration. When running in this mode, the DFSMS/MVS system can share SCDSs, ACDSs and COMMDSs with other systems running MVS/DFP or DFSMS/MVS releases prior to DFSMS/MVS 1.3, and with other DFSMS/MVS systems running in compatibility mode.

compress. (1) To reduce the amount of storage required for a given data set by having the system replace identical words or phrases with a shorter token associated with the word or phrase. (2) To reclaim the unused and unavailable space in a partitioned data set that results from deleting or modifying members by moving all unused space to the end of the data set.

compressed format. A particular type of extended-format data set specified with the (COMPACTION) parameter of data class. VSAM can compress individual records in a compressed-format data set. SAM can compress individual blocks in a compressed-format data set. See *compress*.

concurrent copy. A function to increase the accessibility of data by enabling you to make a consistent backup or copy of data concurrent with the usual application program processing.

connectivity. (1) The considerations regarding how storage controls are joined to DASD and processors to achieve adequate data paths (and alternative data paths) to meet data availability needs. (2) In a DFSMS environment, the system status of volumes and storage groups.

construct. One of the following: data class, storage class, management class, storage group, aggregate group, base configuration.

control data set (CDS). With respect to SMS, a VSAM linear data set containing configurational, operational, or communication information. SMS introduces three types of control data sets: the source control data set, the active control data set, and the communications data set.

CONVERT. A physical volume status indicating that all of the data sets on a volume have an associated storage class and are cataloged in an integrated catalog facility catalog. SMS can select a CONVERT volume for all supported functions. CONVERT volumes are sometimes referred to as converted. See INITIAL.

convert in place. See *in-place conversion*.

coupling facility (CF). The hardware that provides high-speed caching, list processing, and locking functions in a Parallel Sysplex.

coupling facility (CF) cache structure. The CF hardware that provides a data cache.

coupling facility (CF) lock structure. The CF hardware that supports sysplex-wide locking.

cross-system coupling facility (XCF). A component of MVS that provides functions to support cooperation between authorized programs running within a Parallel Sysplex.

D

dasd fast write. An extended function of some models of the IBM 3990 Storage Control in which data is written concurrently to cache and novolatile storage and automatically scheduled for destaging to DASD. Both copies are retained in the storage control until the data is completely written to the DASD, providing data integrity equivalent to writing directly to the DASD. Use of DASD fast write for system-managed data sets is controlled by storage class attributes to improve performance. See also *dynamic cache management*. Contrast with *cache fast write*.

DASD volume. A DASD space identified by a common label and accessed by a set of related addresses. See also *volume*, *primary storage*, *migration level 1*, *migration level 2*.

data class. A collection of allocation and space attributes, defined by the storage administrator, that are used to create a data set.

data collection application. An ISMF application that allows the storage administrator interactively to submit IDCAMS DCOLLECT background jobs or save JCL for later use.

Data Facility Sort (DFSORT). An IBM licensed program that is a high-speed data processing utility. DFSORT provides an efficient and flexible way to handle sorting, merging, and copying operations, as well as providing versatile data manipulation at the record, field, and bit level.

data set. In DFSMS/MVS, the major unit of data storage and retrieval, consisting of a collection of data in one of several prescribed arrangements and described by control information to which the system has access. In OS/390 non-UNIX environments, the terms *data set* and *file* are generally equivalent and sometimes are used interchangeably. See also *file*. In OS/390 UNIX environments, the terms *data set* and *file* have quite distinct meanings.

data set collection. A group of data sets which are intended to be allocated on the same tape volume or set of tape volumes as a result of data set stacking.

data set stacking. The function used to place several data sets on the same tape volume or set of tape volumes. It increases the efficiency of tape media usage and reduces the overall number of tape volumes needed by allocation. It also allows an installation to group related data sets together on a minimum number of tape volumes, which is useful when sending data offsite.

default device geometry. Part of the SMS base configuration, it identifies the number of bytes per track and the number of tracks per cylinder for converting space requests made in tracks or cylinders into bytes, when no unit name has been specified.

default management class. Part of the SMS base configuration, it identifies the management class that should be used for system-managed data sets that do not have a management class assigned.

default unit. Part of the SMS base configuration, it identifies an esoteric (such as SYSDA) or generic (such as 3390) device name. If a user omits the UNIT parameter on the JCL or the dynamic allocation equivalent, SMS applies the default unit if the data set has a disposition of MOD or NEW and is *not* system-managed.

device category. A storage device classification used by SMS. The device categories are as follows SMS-managed DASD, SMS-managed tape, non-SMS-managed DASD non-SMS-managed tape.

Device Support Facilities (ICKDSF). A program used for initialization of DASD volumes and track recovery.

DFSMS environment. An environment that helps automate and centralize the management of storage. This is achieved through a combination of hardware, software, and policies. In the DFSMS environment for MVS, the function is provided by DFSORT, RACF, and the combination of DFSMS/MVS and MVS.

DFSMSdfp. A DFSMS/MVS functional component or base element of OS/390, that provides functions for storage management, data management, program management, device management, and distributed data access.

DFSMSdss. A DFSMS/MVS functional component or base element of OS/390, used to copy, move, dump, and restore data sets and volumes.

DFSMShsm. A DFSMS/MVS functional component or base element of OS/390, used for backing up and recovering data, and managing space on volumes in the storage hierarchy.

DFSMShsm control data set. In DFSMShsm, one of three VSAM key-sequenced data sets that contain records used in DFSMShsm processing. See also *backup control data set*, *migration control data set*, and *offline control data set*.

DFSMS/MVS. An IBM System/390 licensed program that provides storage, data, and device management functions. When combined with MVS/ESA SP Version 5 it composes the base MVS/ESA operating environment. DFSMS/MVS consists of DFSMSdfp, DFSMSdss, DFSMShsm, and DFSMSrmm.

| **DFSMS/MVS Network File System.** See *OS/390*
| *Network File System*.

DFSMSrmm. A DFSMS/MVS functional component or base element of OS/390, that manages removable media.

direct access device space management (DADSM). A collection of subroutines that manages space on disk volumes. The subroutines are: Create, Scratch, Extend, and Partial Release.

DISALL (disable all). Relationship that prevents a system from allocating or accessing data sets in a VIO storage group, a pool storage group, object or object backup storage group, or individual volumes within a pool storage group.

DISNEW (disable new). Relationship that prevents a system from allocating new data sets in a VIO storage group, a pool storage group, object or object backup storage group, or individual volumes within a pool storage group.

Distributed FileManager/MVS. (1) The term used to describe the SAA architectures and programming support that provide distributed file access capabilities between SAA systems. (2) The DFSMS/MVS component that implements the DDM target server.

drive definition. A set of attributes used to define an optical disk drive as a member of a real optical library or pseudo optical library.

dual copy. A high availability function made possible by nonvolatile storage in some models of the IBM 3990 Storage Control. Dual copy maintains two functionally identical copies of designated DASD volumes in the logical 3990 subsystem, and automatically updates both copies every time a write operation is issued to the dual copy logical volume.

dummy storage group. A type of storage group that contains the serial numbers of volumes no longer connected to a system. Dummy storage groups allow existing JCL to function without having to be changed. See also *storage group*.

dump class. A set of characteristics that describes how volume dumps are managed by DFSMShsm.

duplexing. The process of writing two sets of identical records in order to create a second copy of data.

dynamic cache management. A function that automatically determines which data sets will be cached based on the 3990 subsystem load, the characteristics of the data set, and the performance requirements defined by the storage administrator.

E

ENABLE. Relationship that allows a system to allocate and access data sets in a VIO storage group, a pool storage group, or individual volumes within a pool storage group.

expiration. The process by which data sets or objects are identified for deletion because their expiration date or retention period has passed. On DASD, data sets and objects are deleted. On tape, when all data sets have reached their expiration date, the tape volume is available for reuse.

| **extended addressability.** The ability to create and
| access a VSAM data set that is greater than 4 GB in
| size. Extended addressability data sets must be allocated with DSNTYPE=EXT and EXTENDED ADDRESSABILITY=Y.

extended format. The format of a data set that has a data set name type (DSNTYPE) of EXTENDED. The data set is structured logically the same as a data set that is not in extended format but the physical format is different. See also *striped data set* and *compressed format*.

extended remote copy. Extended Remote Copy (XRC) is a technique involving both the DFSMS/MVS host and the I/O Subsystem that keeps a ″real time″ copy of designated data at another location. Updates to the primary center are replicated at the secondary center asynchronously.

F

file. A collection of information treated as a unit. In
| OS/390 non-UNIX environments, the terms *data set* and *file* are generally equivalent and are sometimes used interchangeably. See also *data set*.

file system. In the OS/390 UNIX HFS environment, the collection of files and file management structures on a physical or logical mass storage device, such as a diskette or minidisk. See also *HFS data set*.

| **giga (G).** The information-industry meaning depends upon the context:

1. G = 1,073,741,824(2^{30}) for real and virtual storage
2. G = 1,000,000,000 for disk storage capacity (e.g. 4 GB fixed disk)
3. G = 1,000,000,000 for transmission rates

| **global resource serialization (GRS).** A component of MVS used for serializing use of system resources and for converting hardware reserves on DASD volumes to data set enqueues.

group. (1) With respect to partitioned data sets, a member and the member's aliases that exist in a PDS or PDSE, or in an unloaded PDSE. (2) A collection of users who can share access authorities for protected resources.

guaranteed space. A storage class attribute indicating that space is to be preallocated when a data set is allocated. If explicit volume serial numbers are specified, SMS honors them. If space to satisfy the allocation is not available on the user-specified volumes, the allocation fails.

H

hardware configuration definition (HCD). An interactive interface in MVS that enables an installation to define hardware configurations from a single point of control.

| **HSM complex (HSMplex).** One or more MVS images running DFSMShsm that share a common set of control data sets (MCDS, BCDS, OCDS, and Journal).

hierarchical file system (HFS) data set. A data set that contains a POSIX-compliant file system, which is a collection of files and directories organized in a hierarchical structure, that can be accessed using OS/390 UNIX System Services. See also*file system.*

I

improved data recording capability (IDRC). A recording mode that can increase the effective cartridge data capacity and the effective data rate when enabled and used. IDRC is always enabled on the 3490E Magnetic Tape Subsystem.

inactive data. (1) A copy of active data, such as vital records or a backup copy of a data set. Inactive data is never changed, but can be deleted or superseded by another copy. (2) In tape mount management, data that is written once and never used again. The majority of this data is point-in-time backups. (3) Objects infrequently accessed by users and eligible to be moved to the optical library or shelf. Contrast with *active data*.

index. A set of pointers that are logically ordered by the values of a key. Indexes provide quick access to data and can enforce uniqueness on the rows in a DB2 storage table.

indexed VTOC. A volume table of contents with an index that contains a list of data set names and free space information, which allows data sets to be located more efficiently.

INITIAL. A physical volume status indicating that some data sets on a given volume lack an associated storage class or are not cataloged in an integrated catalog facility catalog. An INITIAL volume is only partially converted to SMS. See *CONVERT*.

in-place conversion. The process of bringing a volume and the data sets it contains under the control of SMS without data movement, using DFSMSdss.

installation exit. The means specifically described in an IBM software product's documentation by which an IBM software product may be modified by a customer's system programmers to change or extend the functions of the IBM software product. Such modifications consist of exit routines written to replace one or more existing modules of an IBM software product, or to add one or more modules or subroutines to an IBM software product, for the purpose of modifying (including extending) the functions of the IBM software product.

integrated catalog facility catalog. A catalog that is composed of a basic catalog structure (BCS) and its related volume tables of contents (VTOCs) and VSAM volume data sets (VVDSs). See also *basic catalog structure* and *VSAM volume data set*.

Interactive Storage Management Facility (ISMF). The interactive interface of DFSMS/MVS that allows users and storage administrators access to the storage management functions.

Interactive System Productivity Facility (ISPF). An IBM licensed program used to develop, test, and run application programs interactively. ISPF is the interactive interface for all storage management functions.

interval migration. In DFSMShsm, automatic migration that occurs when a threshold level of occupancy is reached or exceeded on a DFSMShsm-managed volume, during a specified time interval. Data sets are moved from the volume, largest eligible data set first, until the low threshold of occupancy is reached.

invalid. A status indicating that an SCDS or ACDS is in error. Either the SCDS is incomplete, or an ACS routine in the SCDS has assigned an undefined SMS class name or assigned a storage group list containing undefined names. To be complete, an SCDS must contain at least one storage class definition, at least one pool storage group definition containing at least one volume, a storage group selection routine, and base configuration information. An invalid SCDS cannot be activated.

K

kilo (K). The information-industry meaning depends upon the context:

1. K = 1024(2^{10}) for real and virtual storage
2. K = 1000 for disk storage capacity (e.g. 4000 KB fixed disk)
3. K = 1000 for transmission rates

mega (M). The information-industry meaning depends upon the context:

1. M = 1,048,576(2^{20}) for real and virtual storage
2. M = 1,000,000 for disk storage capacity (e.g. 4000 MB fixed disk)
3. M = 1,000,000 for transmission rates

management class. A collection of management attributes, defined by the storage administrator, used to control the release of allocated but unused space; to control the retention, migration, and backup of data sets; to control the retention and backup of aggregate groups, and to control the retention, backup, and class transition of objects.

manual tape library. A set of tape drives defined as a logical unit by the installation together with the set of system-managed volumes which can be mounted on those drives. See also *tape library*. Contrast with *automated tape library.*

MEDIA2. Enhanced Capacity Cartridge System Tape

MEDIA3. High Performance Cartridge Tape

MEDIA4. Extended High Performance Cartridge Tape

migration. The process of moving unused data to lower cost storage in order to make space for high-availability data. If you wish to use the data set, it must be recalled. See also *migration level 1* and migration level 2.

migration control data set (MCDS). In DFSMShsm, a VSAM key-sequenced data set that contains statistics records, control records, user records, records for data sets that have migrated, and records for volumes under migration control of DFSMShsm.

migration level 1. DFSMShsm-owned DASD volumes that contain data sets migrated from primary storage volumes. The data can be compressed. See also *storage hierarchy*. Contrast with *primary storage* and *migration level 2*.

migration level 2. DFSMShsm-owned tape or DASD volumes that contain data sets migrated from primary storage volumes or from migration level 1 volumes. The data can be compressed. See also *storage hierarchy*. Contrast with *primary storage* and *migration level 1*.

MVS configuration program (MVSCP). A single-step, batch program that defines the input/output configuration to MVS.

MVS/ESA. An MVS operating system environment that supports ESA/390.

MVS/ESA SP. An IBM licensed program used to control the MVS operating system. MVS/ESA SP together with DFSMS/MVS compose the base MVS/ESA operating environment. See also *OS/390*.

MVS system. An MVS image together with its associated hardware, which collectively are often referred to simply as a system, or MVS system.

N

NaviQuest. A component of DFSMSdfp for implementing, verifying, and maintaining your DFSMS SMS environment in batch mode. It provides batch testing and reporting capabilities that can be used to automatically create test cases in bulk, run many other storage management tasks in batch mode, and use supplied ACS code fragments as models when creating your own ACS routines.

nonvolatile storage (NVS). Additional random access electronic storage with a backup battery power source, available with an IBM Cache Storage Control, used to retain data during a power outage. Nonvolatile storage, accessible from all storage directors, stores data during DASD fast write and dual copy operations.

NOTCON (not connected). Relationship that indicates a system is defined but has no access to a VIO storage group, a pool storage group, or individual volumes within a pool storage group.

O

object. A named byte stream having no specific format or record orientation.

object access method (OAM). An access method that provides storage, retrieval, and storage hierarchy management for objects and provides storage and retrieval management for tape volumes contained in system-managed libraries.

OAM complex (OAMplex). One or more instances of OAM running on systems that are part of a parallel sysplex. The OAM systems that are part of an OAMplex share a common OAM database in a DB2 data-sharing group.

OAM-managed volumes. Optical or tape volumes controlled by the object access method (OAM).

object backup storage group. A type of storage group that contains optical or tape volumes used for backup copies of objects. See also *storage group*.

object directory tables. A collection of DB2 tables that contain information about the objects that have been stored in an object storage group.

object storage group. A type of storage group that contains objects on DASD, tape, or optical volumes. See also *storage group*.

object storage hierarchy. A hierarchy consisting of objects stored in DB2 table spaces on DASD, on optical or tape volumes that reside in a library, and on optical or tape volumes that reside on a shelf. See also *storage hierarchy*.

object storage tables. A collection of DB2 tables that contain objects.

OpenEdition MVS. See *OS/390 UNIX System Services*

optical disk drive. The mechanism used to seek, read, and write data on an optical disk. An optical disk drive can be operator-accessible, such as the 3995 Optical Library Dataserver, or stand-alone, such as the 9346 or 9347 optical disk drives.

optical library. A storage device that houses optical drives and optical cartridges, and contains a mechanism for moving optical disks between a cartridge storage area and optical disk drives.

optical volume. Storage space on an optical disk, identified by a volume label. See also *volume*.

OS/390. OS/390 is a network computing-ready, integrated operating system consisting of more than 50 base elements and integrated optional features delivered as a configured, tested system. See also *MVS/ESA SP*.

OS/390 Network File System. A base element of OS/390, that allows remote access to MVS host processor data from workstations, personal computers, or any other system on a TCP/IP network that is using client software for the Network File System protocol.

OS/390 UNIX System Services (OS/390 UNIX). The set of functions provided by the SHELL and UTILITIES, kernel, debugger, file system, C/C++ Run-Time Library, Language Environment, and other elements of the OS/390 operating system that allow users to write and run application programs that conform to UNIX standards.

P

partitioned data set (PDS). A data set on direct access storage that is divided into partitions, called members, each of which can contain a program, part of a program, or data.

partitioned data set extended (PDSE). A system-managed data set that contains an indexed directory and members that are similar to the directory and members of partitioned data sets. A PDSE can be used instead of a partitioned data set.

performance. (1) A measurement of the amount of work a product can produce with a given amount of resources. (2) In a DFSMS environment, a measurement of effective data processing speed with respect to objectives set by the storage administrator. Performance is largely determined by throughput, response time, and system availability.

permanent data set. A user-named data set that is normally retained for longer than the duration of a job or interactive session. Contrast with *temporary data set*.

physical storage. With respect to data, the actual space on a storage device that is to contain data.

physical volume status. Relationship between a volume and SMS. Physical volume status indicates if a volume is SMS-managed, if all of its data sets have an associated storage class, and if all of its data sets are cataloged in an integrated catalog facility catalog. A volume can be in one of three states: CONVERT (fully converted), INITIAL (partially converted), or non-SMS (unconverted).

pool storage group. A type of storage group that contains system-managed DASD volumes. Pool storage groups allow groups of volumes to be managed as a single entity. See also *storage group*.

primary data set. When referring to an entire data set collection, the primary data set is the first data set allocated. For individual data sets being stacked, the primary data set is the one in the data set collection that precedes the data set being stacked and is allocated closest to it.

primary space allocation. Amount of space requested by a user for a data set when it is created. Contrast with *secondary space allocation*.

primary storage. A DASD volume available to users for data allocation. The volumes in primary storage are called primary volumes. See also *storage hierarchy*. Contrast with *migration level 1* and *migration level 2*.

primary volume. A volume managed by DFSMShsm containing data sets that are directly accessible to the user

program management. The task of preparing programs for execution, storing the programs, load modules, or program objects in program libraries, and executing them on the operating system.

program object. All or part of a computer program in a form suitable for loading into virtual storage for execution. Program objects are stored in PDSE program libraries and have fewer restrictions than load modules. Program objects are produced by the binder.

pseudo optical library. A set of shelf-resident optical volumes associated with either a stand-alone or an operator-accessible optical disk drive; see also *real optical library*.

Q

QUIALL (quiesce all). Relationship that prevents a system from scheduling jobs that allocate or access data sets in a VIO storage group, a pool storage group, or individual volumes within a pool storage group.

QUINEW (quiesce new). Relationship that prevents a system from scheduling jobs that allocate new data sets or modify existing ones in a VIO storage group, a pool storage group, or individual volumes within a pool storage group.

R

read-only variable. An ACS language variable that contains data set or system-derived information. It can be referenced but not altered in an ACS routine.

read-write variable. An ACS language variable that is assigned a value within an ACS routine. It can be referenced, and each ACS routine assigns a value to its own, unique read-write variable.

real optical library. Physical storage device that houses optical disk drives and optical cartridges, and contains a mechanism for moving optical disks between a cartridge storage area and optical disk drives. Contrast with *pseudo optical library*.

recovery. The process of rebuilding data after it has been damaged or destroyed, often by using a backup copy of the data or by reapplying transactions recorded in a log.

Redundant Array of Independent Disks (RAID). A direct access storage architecture where data is recorded across multiple physical disks with parity separately recorded so that no loss of access to data results from the loss of any one disk in the array.

Resource Access Control Facility (RACF). An IBM-licensed program or a base element of OS/390, that provides for access control by identifying and verifying the users to the system, authorizing access to protected resources, logging the detected unauthorized attempts to enter the system, and logging the detected accesses to protected resources.

Resource Measurement Facility (RMF). An IBM licensed program or optional element of OS/390, that measures selected areas of system activity and presents the data collected in the format of printed reports, system management facilities (SMF) records, or display reports. Use RMF to evaluate system performance and identify reasons for performance problems.

retained lock. A lock protecting transaction updates when a problem delays transaction recovery of the updates. The retained status is cleared when transaction recovery completes.

S

secondary space allocation. Amount of additional space requested by the user for a data set when primary space is full. Contrast with *primary space allocation*.

sequential data striping. A software implementation of a disk array that distributes data sets across multiple volumes to improve performance.

sharing control data set. A VSAM linear data set that contains information DFSMSdfp needs to ensure the integrity of the data sharing environment.

shelf. A place for storing removable media, such as tape and optical volumes, when they are not being written to or read.

shelf location. A single space on a shelf for storage of removable media.

Small Computer System Interface (SCSI). A mechanical, electrical, and functional standard for a small computer input/output bus and command sets for peripheral devices commonly used with small computers.

SMS class. A list of attributes that SMS applies to data sets and objects having similar allocation (data class), performance (storage class), or backup and retention (management class) needs.

SMS complex. A collection of systems or system groups that share a common configuration. All systems in an SMS complex share a common active control data set (ACDS) and a communications data set (COMMDS). The systems or system groups that share the configuration are defined to SMS in the SMS base configuration.

SMS configuration. A configuration base, Storage Management Subsystem class, group, library, and drive definitions, and ACS routines that the Storage

Management Subsystem uses to manage storage. See also *base configuration* and *source control data set*.

SMS control data set. A VSAM linear data set containing configurational, operational, or communications information that guides the execution of the Storage Management Subsystem. See also *source control data set*, *active control data set*, and *communications data set*.

SMS-managed data set. A data set that has been assigned a storage class. Also called *system-managed data set*.

SMS-managed volume. A volume, managed by SMS, that is defined in the active configuration. See *CONVERT*

source control data set (SCDS). A VSAM linear data set containing an SMS configuration. The SMS configuration in an SCDS can be changed and validated using ISMF. See also *active control data set* and *communications data set*.

spill storage group. An SMS storage group used to satisfy allocations which do not fit into the primary storage group.

storage administrator. A person in the data processing center who is responsible for defining, implementing, and maintaining storage management policies.

storage class. A collection of storage attributes that identify performance goals and availability requirements, defined by the storage administrator, used to select a device that can meet those goals and requirements.

storage control. The component in a storage subsystem that handles interaction between processor channel and storage devices, runs channel commands, and controls storage devices.

storage director. In a 3990 Storage Control, a logical entity consisting of one or more physical storage paths in the same storage cluster. In a 3880, a storage director is equivalent to a storage path.

storage group. A collection of storage volumes and attributes, defined by the storage administrator. The collections can be a group of DASD volumes or tape volumes, or a group of DASD, optical, or tape volumes treated as a single object storage hierarchy. See also *VIO storage group, pool storage group, tape storage group, object storage group, object backup storage group*, and *dummy storage group*.

storage group category. A grouping of specific storage groups which contain the same type of data. This concept is analogous to storage pools in a non-system-managed environment.

storage hierarchy. An arrangement of storage devices with different speeds and capacities. The levels of the storage hierarchy include main storage (memory, DASD cache), primary storage (DASD containing uncompressed data), migration level 1 (DASD containing data in a space-saving format), and migration level 2 (tape cartridges containing data in a space-saving format). See also *primary storage*, *migration level 1*, *migration level 2*, and *object storage hierarchy*.

storage management. The activities of data set allocation, placement, monitoring, migration, backup, recall, recovery, and deletion. These can be done either manually or by using automated processes. The Storage Management Subsystem automates these processes for you, while optimizing storage resources. See also *Storage Management Subsystem*.

storage management cycle. An invocation of the OAM Storage Management Component (OSMC). The purpose of the storage management cycle is to ensure that every object scheduled for processing is placed in the correct level of the object storage hierarchy (as specified by its storage class), is expired or backed up (as specified by its management class or by an explicit application request), and, if necessary, is flagged for action during a subsequent storage management cycle.

Storage Management Subsystem (SMS). A DFSMS/MVS facility used to automate and centralize the management of storage. Using SMS, a storage administrator describes data allocation characteristics, performance and availability goals, backup and retention requirements, and storage requirements to the system through data class, storage class, management class, storage group, and ACS routine definitions.

storage subsystem. A storage control and its attached storage devices. See also *tape subsystem.*

stripe. In DFSMS/MVS, the portion of a striped data set that resides on one volume. The records in that portion are not always logically consecutive. The system distributes records among the stripes such that the volumes can be read from or written to simultaneously to gain better performance. Whether it is striped is not apparent to the application program.

striped data set. In DFSMS/MVS, an extended-format data set consisting of two or more stripes. SMS determines the number of stripes to use based on the value of the SUSTAINED DATA RATE in the storage class. Striped data sets can take advantage of the sequential data striping access technique. See *striping* and *stripe*.

striping. A software implementation of a disk array that distributes a data set across multiple volumes to improve performance.

synchronization time interval. The number of seconds that SMS allows before it checks the

COMMDS for volume status, space statistics, and configurational changes from other systems in the complex.

system data. The data sets required by MVS or its subsystems for initialization and control.

system group. All systems that are part of the same Parallel Sysplex and are running the Storage Management Subsystem with the same configuration, minus any systems in the Parallel Sysplex that are explicitly defined in the SMS configuration.

system-managed buffering for VSAM. A facility available for system-managed extended-format VSAM data sets in which DFSMSdfp determines the type of buffer management technique along with the number of buffers to use, based on data set and application specifications.

system-managed data set. A data set that has been assigned a storage class.

system-managed storage. Storage managed by the Storage Management Subsystem. SMS attempts to deliver required services for availability, performance, and space to applications. See also *DFSMS environment*.

system-managed tape library. A collection of tape volumes and tape devices, defined in the tape configuration database. A system-managed tape library can be automated or manual. See also *tape library*.

system-managed volume. A DASD, optical, or tape volume that belongs to a storage group. Contrast with *DFSMShsm-managed volume* and *DFSMSrmm-managed volume*.

system management facilities (SMF). A component of MVS that collects input/output (I/O) statistics, provided at the data set and storage class levels, which helps you monitor the performance of the direct access storage subsystem.

system programmer. A programmer who plans, generates, maintains, extends, and controls the use of an operating system and applications with the aim of improving overall productivity of an installation.

system residence (SYSRES) volume. The volume on which the nucleus of the operating system and the master catalog are stored.

T

tape configuration database. One or more volume catalogs used to maintain records of system-managed tape libraries and tape volumes.

tape library. A set of equipment and facilities that support an installation's tape environment. This can include tape storage racks, a set of tape drives, and a set of related tape volumes mounted on those drives. See also *system-managed tape library* and *automated tape library*.

tape storage group. A type of storage group that contains system-managed private tape volumes. The tape storage group definition specifies the system-managed tape libraries that can contain tape volumes. See also *storage group*.

temporary data set. An uncataloged data set whose name begins with & or &&, that is normally used only for the duration of a job or interactive session. Contrast with *permanent data set*.

threshold. A storage group attribute that controls the space usage on DASD volumes, as a percentage of occupied tracks versus total tracks. The *low migration threshold* is used during primary space management and interval migration to determine when to stop processing data. The *high allocation threshold* is used to determine candidate volumes for new data set allocations. Volumes with occupancy lower than the high threshold are selected over volumes that meet or exceed the high threshold value.

translate. To check ACS routine source code for syntactic and semantic errors. If no errors exist, the translation process generates an object table from the source code and places the object table into a specified SCDS.

U

unit affinity. Requests that the system allocate different data sets residing on different removable volumes to the same device during execution of the step to reduce the total number of tape drives required to execute the step. Explicit unit affinity is specified by coding the UNIT=AFF JCL keyword on a DD statement. Implicit unit affinity exists when a DD statement requests more volumes than devices.

use attribute. (1) The attribute assigned to a DASD volume that controls when the volume can be used to allocate new data sets; use attributes are *public*, *private*, and *storage*. (2) For system-managed tape volumes, use attributes are *scratch* and *private*.

user group. A group of users in an installation who represent a single department or function within the organization.

V

validate. To check the completeness and consistency of an individual ACS routine or an entire SMS configuration.

virtual input/output (VIO) storage group. A type of storage group that allocates data sets to paging storage, which simulates a DASD volume. VIO storage groups do not contain any actual DASD volumes. See also *storage group*.

vital records. A data set or volume maintained for meeting an externally-imposed retention requirement, such as a legal requirement. Compare with *disaster recovery*.

volume. The storage space on DASD, tape, or optical devices, which is identified by a volume label. See also *DASD volume*, *optical volume*, and *tape volume*.

volume mount analyzer. A program that helps you analyze your current tape environment. With tape mount management, you can identify data sets that can be redirected to the DASD buffer for management using SMS facilities.

volume status. In the Storage Management Subsystem, indicates whether the volume is fully available for system management:

- "Initial" indicates that the volume is not ready for system management because it contains data sets that are ineligible for system management.
- "Converted" indicates that all of the data sets on a volume have an associated storage class and are cataloged in an integrated catalog facility catalog.
- "Non-system-managed" indicates that the volume does not contain any system-managed data sets and has not been initialized as system-managed.

VSAM record-level sharing (VSAM RLS). An extension to VSAM that provides direct record-level sharing of VSAM data sets from multiple address spaces across multiple systems. Record-level sharing uses the System/390 Coupling Facility to provide cross-system locking, local buffer invalidation, and cross-system data caching.

VSAM sphere. The base cluster of a VSAM data set and its associated alternate indexes.

VSAM volume data set (VVDS). A data set that describes the characteristics of VSAM and system-managed data sets residing on a given DASD volume; part of an integrated catalog facility catalog. See also *basic catalog structure* and *integrated catalog facility catalog*.

W

write-once. Describes an optical medium where data can be written one time only.

Numerics

8-name mode. See *compatibility mode*.

32-name mode. For DFSMS/MVS, it is the mode of running SMS on a DFSMS/MVS system in which up to 32 names—representing systems, system groups, or both—are supported in the SMS configuration. When running in this mode, the DFSMS/MVS system can only share SCDSs, ACDSs, and COMMDSs with other DFSMS/MVS systems running in 32-name mode.

Index

Special Characters

% Freespace (data class field) 121
&ACCT_JOB variable 261, 275
&ACCT_STEP variable 261, 275
&ACSENVIR variable 160, 262, 269
&ALLVOL variable 262, 274
&ANYVOL variable 262, 274
&APPLIC variable 263
&DD variable 263
&DEF_DATACLAS variable 263
&DEF_MGMTCLAS variable 263
&DEF_STORCLAS variable 263
&DSN variable 263, 274
&DSNTYPE variable 263
&DSORG variable 264
&DSOWNER variable 264
&DSTYPE variable 264
&EXPDT variable 264
&FILENUM variable 265
&GROUP variable 265
&HLQ variable 265
&JOB variable 265
&LABEL variable 265
&LIBNAME variable 265
&LLQ variable 265
&MAXSIZE variable 265, 269
&MEMHLQ variable 266, 269
&MEMLLQ variable 266, 269
&MEMN variable 266, 269, 274
&MEMNQUAL read-only variable 266
&MEMNQUAL variable 269
&MSDEST read-only variable 266
&MSDEST variable 165
&MSPARM read-only variable 266
&MSPARM variable 165
&MSPOLICY read-only variable 266
&MSPOLICY variable 165
&MSPOOL read-only variable 267
&MSPOOL variable 165
&MSVGP variable 267
&NQUAL variable 267
&NVOL variable 267
&PGM variable 267
&RECORG variable 267
&RETPD variable 267
&SIZE variable 268, 269
&UNIT variable 268, 269
&USER variable 268
&XMODE variable 268

Numerics

32-name support 11, 12

A

ABACKUP 131, 290
Abackup Copy Technique (management class field) 79
ABARS (Aggregate Backup and Recovery Support)
  processing
    pre-DFSMS/MVS aggregate groups 194
    pre-DFSMS/MVS management classes 195
abbreviations 307
Accessibility (storage class field) 93
accessing
  DASD volume 55
  optical volume 61
  pool storage group 47
  remote authorization 299
  VIO storage group 47
ACDS (active control data set)
  allocation 14
  definition 10
  JCL allocation 14
  recovery 209, 210
  relationship with SCDS 10
  saving 12
  share options 15
  size calculation 11
  SMS configuration 10
acronyms 307
ACS (automatic class selection)
  class determination 167
  data set rename 165
  designing and testing ACS routines 139
  DFSMS/MVS NaviQuest 139
  language
    boolean expressions 273
    comparison operators 272
    description 139
    special functions 274
    statements 276
    writing routines 257
  object
    deleting from SCDS 169
    information 168
  read-only variables 228
  read-write variables 259
  routine
    application selection, writing 141
    constants 257
    creating 141
    description 2
    Distributed FileManager/MVS 99, 275
    dummy and SUBSYS data sets 140
    environments 160
    exit for tape management system support 164
    invoking ISPF/PDF editor 142
    language reference 257
    masks 257
    operations 139
    output listing 145
    SCDS validation 149
    storage class example 284
    SYSIN and SYSOUT data sets 140
    testing 151

ACS (automatic class selection) *(continued)*
translating 167
translation listing 145
translation results 144
validating 146
validation results 148
storage group determination 167
variables
read-only 261
read-write 259
VSAM record-level sharing 233
ACTIVATE, ISMF command 172
activate configuration
FIELD resource class 224
RACF authority 224
activating VSAM record-level sharing 246
active control data set 10
Additional Volume Amount (data class field) 121, 201
address space (SMS)
recovery 211
starting 25
Admin or User command Backup (management class field) 76
aggregate group
altering 201
application 131, 132
attributes
altering 202
defining 134
editing 134, 202
backup 137
backup attributes 132
copying SMS 204
defining 132, 137
definition 2
description 131
displaying list 194
instruction data set 132
names 134, 202
panels 132
planning 132
recovering 137
sort criteria 194
view criteria 194
allocation
ACDS 10, 14
COMMDS 14
control data set 9
extended addressability 124
retrying 127
SCDS 13
spreading across multiple volumes 110
storage class, specific volume 97
threshold 53
Allocation/Migration Threshold (storage class field) 53
altering
aggregate group 201
data class 199
management class 197
storage
class 198
altering *(continued)*
storage *(continued)*
group 198
array DASD 93
assigning
data class 129
management class 80
storage class 98
attributes
aggregate group 132, 134, 202
data class
JCL processing 129
record 117, 120
RECORG 129
space 116, 117, 120
VSAM 120, 129
GDG 74
management class
backup 75
expiration 70
migration 72
object class
backup 79
transition 78
point-in-time copy 93
pool storage group 51
retention period 71
storage class
accessibility 93
availability 86, 93
defining for CF cache sets 97, 244
definition 85
storage group 45, 46
Sustained Data Rate (storage class attribute) 92
timing 39
AUDIT parameter, bypass RACF logging 221
authorizing access, VSAM record-level sharing 241
Auto Backup
management class field 77
storage group field 52
Auto Dump (storage group field) 52
Auto Migrate (storage group field) 51
automatic class selection 2
automatic interval migration (storage group field) 51
availability (storage class field) 43
Availability (storage class field) 86, 93
availability, VSAM record-level sharing 233
Avgrec (data class field) 119

B

backup 301
Auto Backup 52
defining aggregate groups 137
management class 75
object copies 59
Backup Control Data Set (BCDS) 303
Backup Copy Technique (management class field) 77
backup devices 94
Backup Frequency (management class field) 75
backup-while-open (BWO) processing. 126
base configuration
active control data set 1

base configuration *(continued)*
adding a system 1
CF cache structures 241
control data set application selection 33
Control Data Set application selection 242
coupling facility definition 33, 242
definition 27
deleting a system 32
description 1
planning SCDS information 27
renaming a system 35
SMS complex 1, 31
VSAM record-level sharing 33, 242
Bias (storage class field) 90
boolean expressions, ACS routine 273
browsing, ACS routine translation 144, 146
buffering, system-managed 124

C

cache
device 88
statistics 184
Cache Fast Write function 297
capacity planning information 301
CF_TIME parameter, IGDSMSxx parmlib member 241, 250
CFRM (coupling facility resource manager) policy 233
channel-to-channel adapter device number 137
CIsize Data (data class field) 121
class transition attributes 66
collection
assigning 42
definition 41
multiple 42
name
OAM 41
qualifiers 267
OAM object 61
command summary, ISMF 287
COMMDS (communications data set)
allocation 14
control data set sizing 11
current utilization statistics 11
definition 11
JCL allocation 14
recovery 209, 210
size calculation 12
SMS complex 11
communications data set 11
Compaction (data class field) 122
comparison operators, ACS routine 272
compatibility mode 10, 28
COMPRESS, IGDSMSxx parameter 17
concurrent copy 93
connection
storage group
to DASD volume 55
to optical volume 61
system to storage group 47, 232
constants, ACS routines 257
constructs
SMS configuration 1
constructs *(continued)*
types 1
continuous availability (storage class field) 43
control data set
allocating 9
application selection panel 27, 33
background 4
description 9
fixed data fields 11
ISMF primary option menu 3
multivolume 13
RACF 226
recovery 209
sharing 9, 233
sizing formula 11
structure changes since DFSMS/MVS 1.3 9
convert status for a DASD volume 39
converting volumes to SMS 208
CONVERTV command, DFSMSdss 159, 208, 224
copying SMS classes, storage and aggregate groups 204
coupling facility (CF)
cache and lock structures 233, 235
cache structures 249
defining cache structures 241
displaying information 247
lock structure 237, 250
monitoring 247
recovery 253
requirements for VSAM record-level sharing 230
resource manager (CFRM) policy 233
statistics monitoring 252
varying cache structures 252
Cycle End Time (storage group field) 57
Cycle Start Time (storage group field) 57

D

DASD (direct access storage device)
fast write function 88
Fast Write function 297
storage
control unit 296
groups 38, 56
object 39
volume
access by storage group 55
data set space 268
deleting from the system 205
status 38, 39
DASDVOL alter authority, RACF 299
data class 2
altering 199
assigning 129
attributes
description 116
JCL processing 129
Recorg 129
creating additional 130
defining
application selection 116
record and space attributes 117, 120

data class 116 *(continued)*
volume 199, 129
deleting 207
description 115
list 191
order when processing attributes 129
planning 115
record and space attributes 117
specifying sort criteria 191
view 191
volume attributes 120
VSAM attributes 120, 129
data collection 301
data set
allocating space 31
allocation 268
extended addressability 124
instruction 132
mask 257
name qualifiers 267
name type (data class) 123
non-recoverable 233
open for VSAM RLS 247
organization
DSORG value 117
qualifiers 274
read-only variables on rename 270
recall processing 166
record organization (RECORG) 267
recover processing 166
recoverable 233
renaming 270
selection 132
data set placement
primary list 105
secondary list 105
tertiary list 105
data set stacking
description 162
read-write variables 260
using
VOL=REF 163
VOL=SER 163
DB2 object storage table 39
DCOLLECT command
active
data set 302
volumes 302
backup data 302
capacity planning data 302
description 301
ISMF panels 301
migration data 302
pool storage groups 304
storage groups 304
volumes 304
DEADLOCK_DETECTION parameter, IGDSMSxx parmlib member 241, 250
default device geometry 31
default management class
SCDS base define field 30
default management class *(continued)*
specifying 30
DEFAULT MANAGEMENT CLASS
specifying 150
Default Unit (SCDS base define field) 31
defining CF cache and lock structures 235
deleting
DASD volumes from the system 205
SMS classes 207
storage groups 206
destaging 298
device status 189
DEVSERV command 189
DFSMS/MVS NaviQuest 139
maintaining SMS configuration 179
storage management tasks in batch 2
DFSMS/MVS Network File System, ACS routines 140
Directory (data class field) 119
DISALL
storage group to optical volume 61
system to storage group 48
DISNEW
storage group to optical volume 61
system to storage group 48
DISPLAY SMS command
cache statistics 184
coupling facility (CF) information 247
OAM status 179
SMS maintenance 181, 189, 190
storage group status 181
trace information 180
DISPLAY XCF command 247
Distributed FileManager/MVS
ACS routines
determining SMS classes 99, 140
using ACS routine variables 275
DO statement 280
drive (optical)
displaying list 194
sort criteria 194
view criteria 194
Drive Start Threshold (storage group field) 57, 59
DSTACK, SMS data set stacking service 18, 174
dual copy 43, 93
dummy storage group 45
dump
Auto Dump 52
classes 52
Dump System Name 52
Dump Class (storage group field) 53
duplexing 298
dynamic data set cache management 88

E

Edit a Data Set (aggregate group field) 135, 203
enable
storage group to optical volume 61
ENABLE
system to storage group 48
END statement
ACS routine
description 284

exit, pre-ACS routine 164
EXIT statement 283
Expdt (data class field) 120
expiration
  date override 20
  EXPDT 70
  management class attributes 70
  object class
    backup attributes 79
    transition attributes 78
  retention limit 70
extended addressability 124
extended format 123

F

fallback, VSAM record-level sharing 230, 254
FALLBACK command 256
false lock contention 239
FIELD resource class 226
FILTLIST statement 278
fixed data fields, sizing 11
flowchart, volume selection 101
formula
  CF lock structure 237, 240
  SHCDS size 234

G

GDG (generation data group)
  management attributes 74
  storage management 67
global connectivity, examples 232
Guaranteed Backup Frequency (storage group attribute) 53
Guaranteed Space (storage class attribute) 95
Guaranteed Synchronous Write (storage class attribute) 97

H

hardware requirements, VSAM RLS 230
HCD (hardware configuration definition) 47
hiperspace size 20

I

ICKDSF (Device Support Facilities) 38
IEFSSNxx, SMS to MVS 23
IF statement 281
IGDSMSxx
  description 15
  modifying for VSAM RLS 241
IGDSSIIN
  optional keyword 24
  SMS initialization 171
IGWLOCK00, CF lock structure 237
Imbed (data class attribute) 121
Initial Access Response Seconds (storage class field) 91
initial status for a DASD volume 39
initializing SYS1.PARMLIB 15
installation, relationship among SCDSs and ACDSs 10
instruction data set
  aggregate group field 135, 203
  definition 132
interval migration, automatic 51
IPL (initial program load) 171
ISMF (Interactive Storage Management Facility)
  ACTIVATE command 172
  application module names 215
  command 287
  description 2
  displaying CF information 248
  function module names 215
  line operator 217
  line operator module names 217
  module names
    line operators 217
    locating 215
  primary option menu 3
  RACF interactions 214
  storage administrator functions 2
  storage group status 184
ISPF (Interactive System Productivity Facility)
  authorizing access 227
  ISMF primary options menu 3
  PDF editor 142
IXCL1DSU Couple Data Set format utility 237

K

key ranges 95
Keylen (data class field) 119
Keyoff (data class field) 119

L

level 0 storage 72
library
  displaying list 194
  names, defining 57, 59
  sort criteria 194
  view criteria 194
Library Names (storage group field) 57, 59
line operators
  summary 290
LIST command 190
listing
  aggregate group 194
  data class 191
  management class 194
  optical
    drive 194
    library 194
  storage
    class 194
    group 194
  tape data class 194
  volumes in a storage group 64
LISTSYS line operator 184
LISTVOL line operator
  displaying with STORGRP parameter 181

LISTVOL line operator *(continued)*
in a storage group
tape 42
listing volumes
DASD 64
mountable optical volume list 64
object and object backup 64
OAM status 179
storage group
DASD 39
lock structure
sharing control 233
lock structure, CF
adjusting size 240
defining 237
false contention 239
retained locks 238
LOG parameter, DEFINE CLUSTER command 247
logstream ID 127
Lrecl (data class field) 119

M

management class
altering 197
application selection 67, 70
assigning 80
attributes
backup 79
defining 67, 68
expiration 70
GDG 74
migration 72
retention limit 71
backup
attributes 75
concurrent copy 77
copy technique 77
frequency 75
version retention period 76
default 27, 30
defining base configuration information 27
deleting 207
description 1, 65, 67
displaying list 194
Management Class field 77
NOACTION 27
planning 66
RACF 226
retention limit 70
sort criteria 194
view criteria 194
Management Class Name (aggregate group field) 134, 202
MANAGEMENTCLASS, access method services command 80
masks, ACS language 257
MAXSYSTEM value 237
Media Interchange (data class field) 122
Media Type (data class field) 122
Member Name (aggregate group field) 135, 203
MGMTCLAS syntax
JCL, TSO/E, IDCAMS parameter 80
MGMTCLAS syntax *(continued)*
OSREQ STORE or CHANGE parameter 80
migrated storage 72
migration
Auto Migrate 51
automatic interval 51
data
collection 302
information 301
level 1 storage 72
level 2 storage 72
management class attributes 72
pool storage group 53
threshold 53
Migration Attributes (management class field) 73
Migration Control Data Set (MCDS) 303
millisecond response for storage class 86, 89
mixing devices in a storage group 105
modifying IGDSMSxx 241
module name, ISMF
commands 217
functions 215
line operators 217
monitoring
coupling facility (CF) 247
statistics 252
monitoring for false lock contention 239
MSS (Mass Storage Subsystem) group name 267
MVS (Multiple Virtual Storage)
command for displaying CF information 247
defining SMS through IEFSSNxx 23

N

naming
SHCDS (sharing control data set) 234
SMS classes 2
NaviQuest
designing and testing ACS routines 139
maintaining SMS configuration 179
storage management tasks in batch 2
NOTCON
storage group to optical volume 61
system to storage group 48
Number of Backup Versions (management class field) 76
Number of Copies (aggregate group field) 134, 202
numeric read-only ACS variables 273

O

OAM (Object Access Method)
activating SCDSs 178
backup storage group 39
cataloging 41
class transition attributes 66
collection names 41
DISPLAY SMS command 179
management class definition 66
object collection 61
restarting 178
status 179, 183

OAM (Object Access Method) *(continued)*
storage group
hierarchy 39
object collection 44
OAMXCF parameter, DISPLAY SMS command 183
object
access method 179
ACS information
deleting 169
displaying 168
backup storage group 39, 46, 60
cataloging 41
character strings 257
class
ACS routines 257
backup attributes 79
transition attributes 78
collection 41, 42
DB2 directory 41
mask representation 257
name 266
reading 40
status field 60
storage group
access types 61
ACS routines 257
attributes 56, 59
backup 46
defining 56
definition 39
devices 61
hierarchy 39, 46
qualifier 39
SCDS 63
timing attributes 39
write requests 59
writing 40
optical
library
names 57, 59
pseudo 40
sharing 40
volume
access 61
label 40
table of contents 40
OSMC (OAM storage management component)
cycle 39
storage management cycle 66
timing attributes 39
OSMC Processing System (storage group field) 57
OSREQ STORE macro 70
Output Data Set Name Prefix (aggregate group field) 134

P

Parallel Sysplex
displaying OAM status 183
MVS environment 5, 31
VSAM record-level sharing 233
Partial Release (management class field) 73, 201
planning
aggregate groups 132
base configuration information 27
data class 115
storage
class 84
group 42
point-in-time copy 93
pool storage group 45
pre-ACS routine exit 164
primary
location 132
option menu 25
storage 72
primary list, SMS volume selection 106
PROC statement 277
program control feature 214
pseudo optical library 40

Q

Qualifier field (object storage group) 57
QUIALL, SMS status 48
QUINEW, SMS status 48

R

RACF (Resource Access Control Facility)
authorization structure 215
bypass logging 221
DASDVOL alter authority 299
FACILITY profiles for storage administration 221
FIELD resource class 226
ISMF functions 214
program control feature 214
protecting modules 220
RDEFINE command 220, 221
resource owner 214
RAUTH line operator 299
RDEFINE command 220
read caching 297
read-only variables
ACS routine 272
pre-ACS routine exit 165
restrictions 271
recall processing 166
Recfm record format for data class field 118
record, data class attributes 117
Recording Technology (data class field) 122
Recorg (data class field) 118
recover processing 166
recovery
location 132
SMS
address space 211
complex 211
information 209, 211
VSAM record-level sharing 253
reducing false lock contention 240
remote access 299
Replicate (data class field) 121
resource class, FIELD 226

restarting OAM 178
Retain Days Extra Backup Versions (management class field) 76
Retain days only Backup Version (management class field) 76
retained locks 238
Retention Limit (management class field) 70
retention period
  limit 70
  management class comparison 71
  read-only variable 267
Retpd (data class field) 120
RLS_MAX_POOL_SIZE parameter, IGDSMSxx parmlib member 236, 241, 251
RLSINIT parameter, IGDSMSxx parmlib member 241, 251
rolled-off GDS 74

S

SCDS (source control data set) 10
  ACS routine validation 149
  allocation 13
  base configuration definition 27, 33
  changing SMS configurations 177
  copying object storage group 63
  coupling facility definition 33, 242
  defining status 28
  definition 10
  JCL allocation 13
  recovery 209
  relationship with ACDS 10
  saving 12
  size calculation 11
  SMS configuration 9
SCDS Status field (SCDS base define) 30
secondary list, SMS volume selection 106
SELECT statement 281
selection data set
  aggregate group field 135, 203
  description 132
sequential data striping
  batch processing 92
  sustained data rate (SDR) 92
SET SMS command 25
SET statement 279
SETCACHE line operator 296
SETSMS operator command 172, 173
SETXCF START command 239, 249
Shareoptions parameter
  data class field 121
SHAREOPTIONS parameter
  specified for an ACDS 15
  specified for an SHCDS 234
sharing
  control data sets 9
sharing control data set (SHCDS) 233
SHCDS (sharing control data set) 233, 254
  share options 234
size
  CF cache structure 235
  CF lock structure 237
size *(continued)*
  SHCDS (sharing control data set) 235
SMF_TIME parameter, IGDSMSxx parmlib member 241, 251
SMS (Storage Management Subsystem)
  32-name support 11, 12
  active
    configuration information 179
    data set information 301
    volume information 301
  address space 25, 211
  base configuration 1
  cache statistics 184
  capacity planning information 301
  classes, determining 167
  complex 1, 5, 31
  complex recovery 211
  configuration
    activation 4
    automatic activation 172
    changing parameters 173
    content 1
    manual activation 171
    minimal 149
  control data set, allocating 4
  converting volumes to SMS 208
  copying classes, storage groups, aggregate groups 204
  default management class 30
  definition 1
  deleting classes 207
  device status 189
  IGDSMSxx 15
  information recovery 209
  initialization parameters 15
  IPL of SMS complex systems 171
  management class, default 30
  modifying SYS1.PARMLIB 15
  MVS with IEFSSNxx 23
  operator commands
    DISPLAY SMS command 179
    VARY SMS command 179
  Parallel Sysplex 5, 31
  physical storage 37
  preliminary steps 9
  RACF
    function 214
    STGADMIN profiles 221
  record definition 23
  resource owner 214
  storage group
    deleting 205
    detail status 183
    determining 167
    LISTSYS line operator 184
    moving volumes 196
    redefining volumes 196
    status 181
    volume status 182
  storage group status (storage group field) 60
  Storage Group Status (storage group field) 58

SMS (Storage Management Subsystem) *(continued)*
system
group 5, 31
name 5, 31
TRACE information 180
volume
selection 100
volume status 187
writing the SMS entry 23
SMSVSAM address space 246, 254
SnapShot feature, RAMAC Virtual Array 93
sort criteria
aggregate group 194
data class 191
management class 194
optical
drive 194
library 194
storage
class 194
group 194
space
attributes 117
constraints 127
use information 301
Space (data class field) 119
Space Constraint Relief (data class field) 127
statistics
active configuration 179
cache 184
coupling facility monitoring 252
device 189
OAM 179
storage group
DISPLAY SMS command 181
LISTSYS line operator 184
LISTVOL line operator 181
volume 182
volume 187
STGADMIN profiles
command and keyword related 221
Facility Class Profiles 224
storage administrator
accessing primary option menu 25
authorizing access 227
Control Data Set Application selection 27
control data set application selection 33
Control Data Set Application selection 242
ISMF primary option menu 3
storage class 97
accessibility attributes 93
ACS routine sample 284
altering 198
application selection panel 85, 97, 244
assigning 98
attributes 85, 97, 244
Availability attributes 86
availability attributes 93
bias 90
copying SMS 204
coupling facility (CF) 84
storage class 105 *(continued)*
data set placement 93
defining additional 99
definition 85, 244
deleting 207
description 1, 83
device failure 93
displaying list 194
dual copy 93
guaranteed space 95
initial access response seconds 91
millisecond response 86, 89, 90
performance 86
planning 84
point-in-time copy attributes 93
protection 226
RACF 226
sort criteria 194
tape data sets 84
tape volumes 84
view criteria 194
volume allocation 97
VSAM record-level sharing 97, 245
storage control unit 296
storage group
ACS routine 167
additional groups 63
altering 196
application selection panel 44
assigning 61
attributes 44, 46
Auto Migrate 51
automatic interval migration 51
backup object attributes 59
changing 196
convert status 39
copying SMS 204
DASD volume
relationship to systems in SMS complex 55
status 38
defining volume status 55
definition 37, 44
deleting 205
description 1
displaying list 194
dummy
attributes 45, 49
volume serial numbers 50
dump classes 53
LIST commands 190
location 39
migration threshold 53
multiple 40
OAM collection names 41
object
attributes 56
backup 46
directory 44
storage hierarchy 39
optical volumes 61
planning for OAM 42, 44

storage group *(continued)*
pool 167, 51
qualifier 39
sort criteria 194
status
DISPLAY SMS command 181
LISTSYS line operator 184
volume 55
Status field 54, 62
storage panels 46
system access 47
tape
defining 62
library definition 42
SMS support 42
types 37, 39
view criteria 194
VIO 45
volume
characteristics 38
moving 196
redefining 196
status 182
Storage Group Name (storage group field) 45
Storage Group Type (storage group field) 45
Storage Management Subsystem 1
striping volume selection 110
sustained data rate (SDR) 92
SYS1.PARMLIB data set
IEASYSyy member 15, 23
IEFSSNxx member 15, 23
IGDSMSxx member 15, 241
system group, system name 31
System Name 52

T

T SMS command 25, 171
tape management system support 164
tape mount management 52, 53
tape storage group 46
tertiary list, SMS volume selection 107
testing
ACS routine 151
running ACS routine cases 155
translation
ACS routine
description 142
running with mixed levels of DFSMS/MVS 150
TSO (time sharing option), authorizing access 227

V

validation
ACS routine
conditions 147
entire SCDS 149
running with mixed levels of DFSMS/MVS 150
browsing ACS routine results 148
individual ACS routine 146
variables, read-only
ACS routine 261
variables, read-only *(continued)*
definition 261
list 261
restrictions 271
variables, read-write
definition 259
list 259
VARY SMS command 252
versioning devices 94
VIEW CRITERIA (data class field) 191
VIO (virtual I/O)
Maxsize (storage group field) 47
storage group allocation 45
VIO Maxsize (storage group field) 47
VIO Unit (storage group field) 47
virtual concurrent copy 93
volume
attributes (data class) 120
data class 120
DISPLAY SMS command 187
dummy storage group 50
selection, enhanced 105
selection (SMS) 100
serial number comparison 274
status 187
storage characteristics 38
VSAM attributes 120
Volume Count (data class field) 121
Volume Full at Write Error (storage group field) 58, 59
Volume Full Threshold (storage group field) 57, 59
VSAM (virtual storage access method)
attributes (data class) 120, 129
data class attributes 120, 129
storage groups 38
VSAM record-level sharing
activating 246
adjusting lock structure size 240
authorization 241
CF cache and lock structures 235
coupling facility (CF) 97, 230
displaying CF information 247
environment 230
fallback processing 230, 254
false lock contention 239
hardware requirements 230
lock structure 237
modifying the SYS1.PARMLIB IGDSMSxx member 241
monitoring the coupling facility (CF) 247
overview 229
planning for availability 233
recovery 253
requirements 246
retained locks 238
SHCDS 233
storage class definition 244
system connectivity requirements 232
VTS (Virtual Tape Server) 266

W

WRITE statement 283
writing SMS entry 23

X

XES coupling definition process 237

Readers' Comments — We'd Like to Hear from You

DFSMS/MVS Version 1 Release 5
DFSMSdfp Storage Administration Reference

Publication No. SC26-4920-05

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

Phone No.

Readers' Comments — We'd Like to Hear from You
SC26-4920-05

Cut or Fold
Along Line

Fold and Tape | **Please do not staple** | Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
RCF Processing Department
G26/050
5600 Cottle Road
SAN JOSE, CA
U.S.A. 95193-0000

Fold and Tape | **Please do not staple** | Fold and Tape

Cut or Fold
Along Line

File Number: S370/S390-34
Program Number: 5695–DF1
5647–A01

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.

SC26-4920-05

Spine information:

IBM

DFSMS/MVS Version 1 Release 5

DFSMS/MVS V1R5 DFSMSdfp Storage Admin. Reference